

1 - Desenvolver uma API com JavaWeb utilizando Servlets ou JAX-RS:

Ele deve permitir realizar a persistência de uma classe **através de um formulário** com os campos que armazenaria informações de uma compra pela internet, como o nome do cliente, endereço (Rua, N, cidade, estado), valor da compra, CEP de destino, valor do frete, prazo de entrega, produto escolhido para entrega entre outras informações que considerar pertinente.

Na aplicação, antes de retornar os dados do valor do frete, o usuário/login precisa ser validado.

Desta forma, deve-se implementar um filter, para processar a informação ou simplesmente retornar um erro informando que é necessário um usuário válido.

Após o filter encaminhar a requisição para o controller, deve-se seguir as implementações:

Ao inserir um novo registro, deve ser realizado o calculo do frete e o prazo da entrega com base no valor de cada estado e seu respectivo prazo.

Exemplo:

Pernambuco – valor do frete: 120,00 → Prazo: 10 dias

Rio Grande do Sul – valor do frete: 60,00 → Prazo: 12 dias

O Servlet 1 retornará um **JSON** com o valor do frete e o prazo de entrega.

```
{
  valorFrete: 120.00,
  prazo: 10
}
```

O Servlet 2 retornará uma página JSP com o valor do frete e o prazo.

Importante:

Estruturar o padrão MVC;

Criar Service para separar a lógica do controlador;

Divisão de pacotes e boas práticas de POO;

Tratamento de exceções;

Enviar link do repositório git;

Utilizar MySQL como banco de dados;

Não aceitaremos a entrega após o prazo.



2 – Criar um Web Service Java com **JAX-RS** ou **Servlets**, que retorne dados de previsão do tempo. A API que será desenvolvida deve receber alguns dados de localização, como cidade e estado. A consulta da previsão é para o dia atual. Estes dados, devem ser tratados simulando uma API de previsão de tempo. Sua API deve retornar um JSON com a previsão do tempo atual.

```
{
  cidade: Olinda,
  temperatura: 45
}
```

Extra: Retornar JSON previsão com tempo dos próximos 5 dias (controlador 1)

Extra 2: Retornar previsão do tempo dos próximos 5 dias em uma página HTML (controlador 2)

```
{
  data: 04/03/2022
  cidade: Olinda,
  temperatura: 32
},
{
  data: 05/03/2022
  cidade: Olinda,
  temperatura: 15
},
{...}
}
```

Importante:

- Estruturar o padrão MVC;
- Criar Service para separar a lógica do controlador;
- Divisão de pacotes e boas práticas de POO;
- Tratamento de exceções;
- Enviar link do repositório git;
- Utilizar MySQL como banco de dados;

Não aceitaremos a entrega após o prazo.

Prazo de entrega:



04/07/2022



13:00



andre.lima@compasso.com.br;
renan.acosta@compasso.com.br;
thauany.martins@compasso.com.br



Everybody
needs a
Compass.