



Cento Universitário UNA

Sistemas de Informação

Recuperação de Informação

Práticas de Laboratório

Wesley Dias Maciel

2019/01



Centro Universitário UNA
Sistemas de Informação
Recuperação de Informação
Prática de Laboratório
Wesley Dias Maciel
2019/01

Spring Boot, Hibernate e JPA



Prática 11

- 1) Você está recebendo, juntamente com esta prática, o projeto da aplicação. Nessa versão, o projeto implementa uma nova forma de consultar URLs sementes. As consultas são realizadas através da anotação `@Query`, executando consultas nativas do banco de dados.

Retornar as URLs sementes em ordem alfabética do campo “url”

No repositório **LinkRepository**, foi criado o método:

```
@Query (value = "SELECT * FROM link ORDER BY url", nativeQuery = true)  
List<Link> getInLexicalOrder ();
```

Esse método foi anotado com a anotação `@Query`. Essa anotação permite especificar a consulta que se deseja realizar no banco de dados. O método retorna todas as URLs sementes em ordem alfabética, com base no campo “url”.

No serviço **ColetorService**, foi criado o método:

```
public List<Link> listarEmOrdemAlfabetica () {  
    return lr.getInLexicalOrder ();  
}
```

No controlador **Coletor**, foi criado o método:

```
// Request for: http://localhost:8080/coletor/link/ordemAlfabetica  
  
@GetMapping (value = "/link/ordemAlfabetica", produces =  
MediaType.APPLICATION_JSON_UTF8_VALUE)  
  
public ResponseEntity listarEmOrdemAlfabetica () {  
    return new ResponseEntity (cs.listarEmOrdemAlfabetica (), HttpStatus.OK);  
}
```

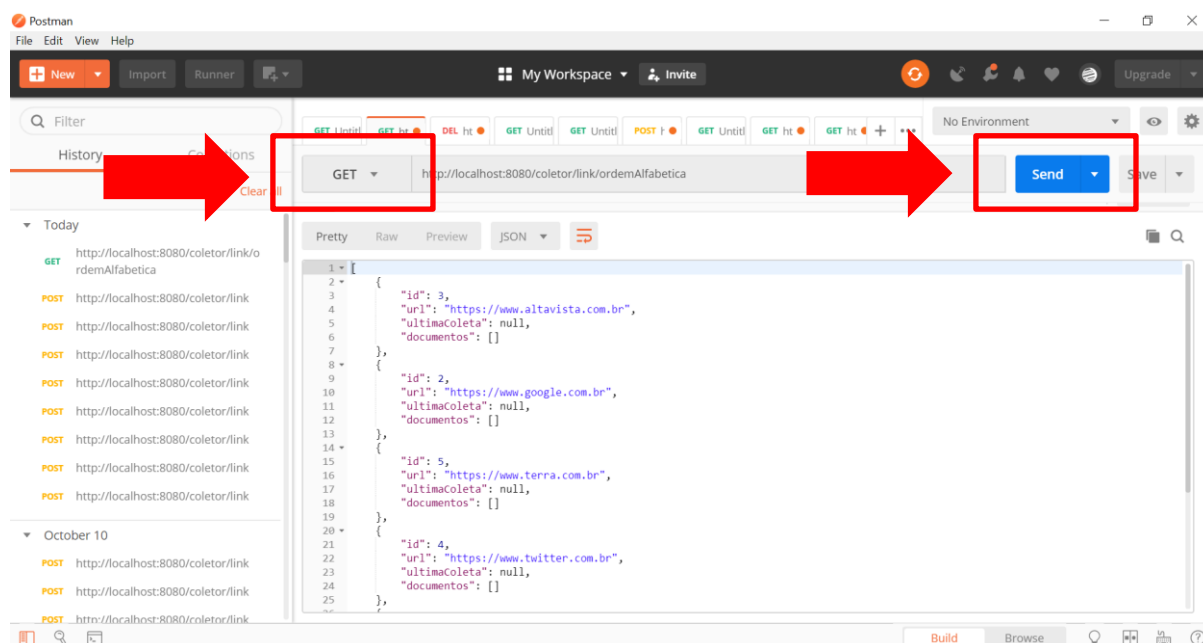


Centro Universitário UNA
Sistemas de Informação
Recuperação de Informação
Prática de Laboratório
Wesley Dias Maciel
2019/01

Para testar a API, use um programa para teste de APIs REST, como o Postman (<https://www.getpostman.com/>).

Execute o projeto e observe as respostas geradas pelo servidor no Postman.

URLs sementes inseridas no banco de dados, listadas em ordem alfabética:



- 2) Em seu projeto, crie a API que permita consultar nomes de administradores, de usuários e de hosts em ordem alfabética. Empregue a anotação `@Query`, para especificação de consultas nativas do banco de dados. Realize a validação de parâmetros recebidos. Para todos os métodos que interagem com os clientes, retorne respostas com mensagens significativas, obedecendo os códigos adequados do protocolo HTTP.

Lista de códigos de status HTTP:

1xx Informativa

- 100 Continuar
- 101 Mudando protocolos
- 102 Processamento (WebDAV) (RFC 2518)
- 122 Pedido-URI muito longo

2xx Sucesso

- 200 OK
- 201 Criado
- 202 Aceito



203 não-autorizado (desde HTTP/1.1)
204 Nenhum conteúdo
205 Reset
206 Conteúdo parcial
207-Status Multi (WebDAV) (RFC 4918)

3xx Redirecionamento

300 Múltipla escolha
301 Movido
302 Encontrado
303 Consulte Outros
304 Não modificado
305 Use Proxy (desde HTTP/1.1)
306 Proxy Switch
307 Redirecionamento temporário (desde HTTP/1.1)
308 Redirecionamento permanente (RFC 7538[2])

4xx Erro de cliente

400 Requisição inválida
401 Não autorizado
402 Pagamento necessário
403 Proibido
404 Não encontrado
405 Método não permitido
406 Não Aceitável
407 Autenticação de proxy necessária
408 Tempo de requisição esgotou (Timeout)
409 Conflito
410 Gone
411 comprimento necessário
412 Pré-condição falhou
413 Entidade de solicitação muito grande
414 Pedido-URI Too Long
415 Tipo de mídia não suportado
416 Solicitada de Faixa Não Satisfatória
417 Falha na expectativa
418 Eu sou um bule de chá
422 Entidade improcessável (WebDAV) (RFC 4918)
423 Fechado (WebDAV) (RFC 4918)
424 Falha de Dependência (WebDAV) (RFC 4918)
425 coleção não ordenada (RFC 3648)
426 Upgrade Obrigatório (RFC 2817)
450 bloqueados pelo Controle de Pais do Windows
499 cliente fechou Pedido (utilizado em ERPs/VPsA)

5xx outros erros (erro de servidor)

500 Erro interno do servidor (Internal Server Error)
501 Não implementado (Not implemented)
502 Bad Gateway
503 Serviço indisponível (Service Unavailable)
504 Gateway Time-Out
505 HTTP Version not supported