

Generation Brasil – Projeto Integrador

Projeto: Plataforma WEB – ColaborArt.

Tipo: E-commerce.

Representantes: Alana Rodrigues Teixeira / Bruno Falango / Gustavo Broch /

Laíse Lima Galvão de Souza / Vitor Carneiro Borela.

Back-End

Framework: Spring Boot

Link: <https://start.spring.io/>

Versão: 2.4.1

Projeto: Maven Project

Linguagem: Java

Versão: 8

Dependências Adicionadas:

- Spring Web
- Spring Boot DevTools
- Spring Data JPA
- Validation
- MySQL Driver

Banco de Dados: MySQL

Model

Categoria - Atributos:

- **private long idCategoria;**

Número de identificação da categoria dentro da *tb_categoria*.

(Exemplo: 1, 2, 3, etc.)

- **private String tipoProduto;**

Nome da categoria que objetiva organizar os produtos.

(Exemplo: Caneca, camisetas, etc.)

- **private String descricaoCategoria;**

Descrição da categoria do produto.

(Exemplo: Material de composição, linha (estilo / coleção), etc.)

Categoria – Métodos:

- **getters / setters dos atributos do model;**

```
public long getIdCategoria() {  
    return idCategoria;  
}
```

```
public void setIdCategoria(long idCategoria) {  
    this.idCategoria = idCategoria;  
}
```

```
public String getTipoProduto() {  
    return tipoProduto;  
}
```

```
public void setTipoProduto(String tipoProduto) {  
    this.tipoProduto = tipoProduto;  
}
```

```
public String getDescricaoCategoria() {  
    return descricaoCategoria;  
}
```

```
public void setDescricaoCategoria(String descricaoCategoria) {  
    this.descricaoCategoria = descricaoCategoria;  
}
```

Repository

```
-  
0 @Repository  
1 public interface CategoriaRepository extends JpaRepository<Categoria, Long> {  
2  
3     public List<Categoria> findByTipoProdutoContainingIgnoreCase(String tipoProduto);  
4 }  
5
```

Interface – CategoriaRepository

Criação do repositório da categoria, estendendo da Classe **JpaRepository**, onde recebe a **Categoria** Model e o tipo primitivo **Long**.

Método – Query Method

- `findByTipoProdutoContainingIgnoreCase`
Lista todo tipo de produto presente na `tb_categoria`, de acordo com a texto passado pelo usuário.

Controller

```
@Autowired
private CategoriaRepository repositorio;
```

Atributo para implementação do repositório, através da injeção de dependências (@Autowired).

```
@GetMapping()
public ResponseEntity<List<Categoria>> buscarTodasCategorias(){
    return ResponseEntity.ok(repositorio.findAll());
}
```

Retorna todas as categorias encontradas no banco de dados.

```
@GetMapping("/{id}")
public ResponseEntity<Categoria> buscarCategoriaPorId(@PathVariable long id){
    return repositorio.findById(id).map(resposta -> ResponseEntity.ok(resposta))
        .orElse(ResponseEntity.notFound().build());
}
```

Retorna a Categoria de acordo com Id fornecido pelo usuário.

```
@GetMapping("/produtos/{tipoProduto}")
public ResponseEntity<List<Categoria>> buscarProdutoPorCategoria(@PathVariable String tipoProduto){
    return ResponseEntity.ok(repositorio.findByTipoProdutoContainingIgnoreCase(tipoProduto));
}
```

Retorna valores da Categoria, de acordo com o tipo do produto fornecido (Query Method).

```
@PostMapping()
public ResponseEntity<Categoria> inserirCategoria(@RequestBody Categoria categoria){
    return ResponseEntity.status(HttpStatus.CREATED).body(repositorio.save(categoria));
}
```

Insere informações da Categoria no banco de dados.

```
@PutMapping()
public ResponseEntity<Categoria> atualizarCategoria(@RequestBody Categoria categoria){
    return ResponseEntity.status(HttpStatus.OK).body(repository.save(categoria));
}
```

Atualiza informações da Categoria no banco de dados, de acordo com o Id.

```
@DeleteMapping("/{id}")
public void deleteCategoria(@PathVariable long id) {
    repository.deleteById(id);
}
```

Apaga informações da Categoria no banco de dados.

CRUD – Postman

GET

<http://localhost:8080/categoria/>

```
{
  "idCategoria": 1,
  "tipoProduto": "Camiseta Gamer",
  "descricaoCategoria": "Camisetas produzidas com material 100% algodão, confortável e fácil de lavar, disponível em PP, P, M, G, GG, EXG e EXG2"
},
{
  "idCategoria": 2,
  "tipoProduto": "Camiseta BodyTattoo",
  "descricaoCategoria": "Camisetas produzidas com material 90% algodão mesmo e 10% poliéster, disponível em PP, P, M, G, GG, EXG e EXG2"
},
]
```

GET

<http://localhost:8080/categoria/1>

```
1 {
2   "idCategoria": 1,
3   "tipoProduto": "Camiseta Gamer",
4   "descricaoCategoria": "Camisetas produzidas com material 100% algodão, confortável e fácil de lavar, disponível em PP, P, M, G, GG, EXG e EXG2"
5 }
```

GET

<http://localhost:8080/categoria/produtos/histórica>

```
{
  "idCategoria": 5,
  "tipoProduto": "Camiseta Histórica",
  "descricaoCategoria": "Camisetas produzidas com material 90% algodão e 10% elastano, disponível em PP, P, M, G, GG, EXG e EXG2"
},
{
  "idCategoria": 6,
  "tipoProduto": "Caneca Histórica",
  "descricaoCategoria": "Caneca Express térmica, disponível em 350ml e 450ml"
}
```

POST

<http://localhost:8080/categoria/>



PUT

<http://localhost:8080/categoria/>



DELETE

<http://localhost:8080/categoria/9>

