

# Documentação do arquivo BoatReviews.cls

## Introdução

O arquivo `BoatReviews.cls` contém uma classe Apex que é usada para recuperar avaliações de barcos (`BoatReview__c`) associadas a um determinado barco (`Boat__c`). Essa classe é projetada para ser usada em um contexto Lightning (Aura ou LWC) e fornece um método estático acessível para recuperar os dados necessários.

## Descrição

A classe `BoatReviews` é uma classe Apex com o modificador `with sharing`, o que significa que ela respeita as regras de compartilhamento do Salesforce. Ela contém um método estático chamado `getAll`, que é anotado com `@AuraEnabled`, permitindo que ele seja chamado a partir de componentes Lightning. O método retorna uma lista de registros do objeto personalizado `BoatReview__c` associados a um barco específico, identificado pelo `Id` fornecido como parâmetro.

## Estrutura

A classe é composta por:

- Uma declaração de classe com o modificador `with sharing`.
- Um método estático público chamado `getAll`, que é anotado com `@AuraEnabled`.

## Dependências

A classe depende dos seguintes elementos:

- Objeto personalizado `BoatReview__c`.
- Campo de relacionamento `Boat__c` no objeto `BoatReview__c`.
- Campos no objeto `BoatReview__c` como `Name`, `Comment__c`, `Rating__c`, entre outros.

## Imports

Não há imports explícitos em Apex, mas a classe utiliza o objeto personalizado `BoatReview__c` e seus campos.

## Variáveis

A classe não possui variáveis declaradas explicitamente. O método `getAll` utiliza variáveis locais implícitas para armazenar os resultados da consulta SOQL.

## Métodos

**`getAll(Id boatId)`**

### Descrição:

Este método recupera todas as avaliações (`BoatReview__c`) associadas a um barco específico (`Boat__c`). Ele utiliza uma consulta SOQL para buscar os registros e retorna uma lista de objetos `BoatReview__c`.

### Parâmetros:

- `boatId` (Id): O identificador do barco cujas avaliações devem ser recuperadas.

### Retorno:

- `List<BoatReview__c>`: Uma lista de registros do objeto `BoatReview__c` contendo informações como nome, comentário, classificação, datas de criação/modificação e informações do criador.

### Anotações:

- `@AuraEnabled`: Permite que o método seja chamado a partir de componentes Lightning.

### Consulta SOQL:

```
SELECT Name, Comment__c, Rating__c, LastModifiedDate, CreatedDate, CreatedBy.Name, CreatedBy.SmallPhotoUrl, CreatedBy.CompanyName
FROM BoatReview__c
WHERE Boat__c = :boatId
```

## Exemplo

Aqui está um exemplo de como usar o método `getAll` em um componente Lightning:

```
import { LightningElement, wire } from 'lwc';
import getAll from '@salesforce/apex/BoatReviews.getAll';

export default class BoatReviewsComponent extends LightningElement {
  @api boatId;
  reviews;
```

```

@wire(getAll, { boatId: '$boatId' })
wiredReviews({ error, data }) {
  if (data) {
    this.reviews = data;
  } else if (error) {
    console.error(error);
  }
}
}
}

```

## Diagrama de Dependência

O diagrama abaixo ilustra as dependências da classe BoatReviews:

```

classDiagram
    class BoatReviews {
        +List~BoatReview__c~ getAll(Id boatId)
    }
    BoatReviews --> BoatReview__c : Consulta SOQL
    BoatReview__c : Name
    BoatReview__c : Comment__c
    BoatReview__c : Rating__c
    BoatReview__c : LastModifiedDate
    BoatReview__c : CreatedDate
    BoatReview__c : CreatedBy.Name
    BoatReview__c : CreatedBy.SmallPhotoUrl
    BoatReview__c : CreatedBy.CompanyName

```

## Notas

- A classe utiliza o modificador `with sharing`, garantindo que as regras de compartilhamento sejam respeitadas ao executar a consulta SOQL.
- O método `getAll` é projetado para ser usado em componentes Lightning, como Aura Components ou Lightning Web Components (LWC).

## Vulnerabilidades

- **Injeção SOQL:** O método utiliza uma variável `bind (:boatId)` na consulta SOQL, o que ajuda a prevenir injeções SOQL. No entanto, é importante garantir que o `boatId` seja validado antes de ser passado para o método.
- **Limites de Governança:** Dependendo do número de registros retornados, a consulta pode atingir os limites de governança do Salesforce. É recomendável implementar paginação se o número de registros for muito grande.