🖥 **arthurgregorio** / **web-budget**   (Public)

<> **Code**   ⊙ Issues **15**   ⇵ **Pull requests** **26**   ▷ Actions   ⊞ Projects **1**   📖 Wiki   ⊘ Security   ⩘ In

Commit

⚠ This commit does not belong to any branch on this repository, and may belong to a fork outside of the repository.

rm de com desnecessario	**Browse files**
👤 **deniserisia** committed on Jun 17	

1 parent `0fc1c72`   commit `519e6aa`

Showing **1 changed file** with **0 additions** and **59 deletions**. | Whitespace | Ignore whitespace   Split | Unified |

⌄ ✛ **59** ■■■■■ src/main/java/br/com/webbudget/domain/entities/financial/PeriodMovement.java ⧉

```
 81   81        @JoinColumn(name = "id_financial_period")
 82   82        @NotNull(message = "{period-movement.financial-period}")
 83   83        private FinancialPeriod financialPeriod;
 84      -
 85      -      /**
 86      -       * Used at the credit card invoice to mark the movement and help the user to check the invoice
 87      -       */
 88   84        @Getter
 89   85        @Setter
 90   86        @Transient
 91   87        private boolean checked;
 92   88
 93      -      /**
 94      -       * Constructor...
 95      -       */
 96   89        public PeriodMovement() {
 97   90            super();
 98   91            this.dueDate = LocalDate.now();
 99   92            this.periodMovementType = PeriodMovementType.MOVEMENT;
100   93            this.periodMovementState = PeriodMovementState.OPEN;
101   94        }
102   95
103      -      /**
104      -       * Get the payment date of this movement
105      -       *
106      -       * @return the payment date
107      -       */
108   96        public LocalDate getPaymentDate() {
109   97            return this.payment != null ? this.payment.getPaidOn() : null;
110   98        }
111   99
112      -      /**
113      -       * To check if this movement is paid with a credit {@link Card}
114      -       *
```

```
115                    -        * @return true if is, false if not
116                    -        */
117      100           public boolean isPaidWithCreditCard() {
118      101               return this.payment != null && this.payment.isPaidWithCreditCard();
119      102           }
120      103
121                    -       /**
122                    -        * To check if this movement is paid with a debit {@link Card}
123                    -        *
124                    -        * @return true if is, false if not
125                    -        */
126      104           public boolean isPaidWithDebitCard() {
127      105               return this.payment != null && this.payment.isPaidWithDebitCard();
128      106           }
129      107
130                    -       /**
131                    -        * To check if this movement is paid with a cash
132                    -        *
133                    -        * @return true if is, false if not
134                    -        */
135      108           public boolean isPaidWithCash() {
136      109               return this.payment != null && this.payment.isPaidWithCash();
137      110           }
138      111
139                    -       /**
140                    -        * To check if this movement is a {@link CreditCardInvoice}
141                    -        *
142                    -        * @return true if is, false otherwise
143                    -        */
144      112           public boolean isCreditCardInvoice() {
145      113               return this.periodMovementType == PeriodMovementType.CARD_INVOICE;
146      114           }
147      115
148                    -       /**
149                    -        * To check if this movement is paid or not
150                    -        *
151                    -        * @return true if paid, false otherwise
152                    -        */
153      116           public boolean isPaid() {
154      117               return this.periodMovementState == PeriodMovementState.PAID;
155      118           }
156                    -
157                    -       /**
158                    -        * To check if this movement is open or not
159                    -        *
160                    -        * @return true if paid, false otherwise
161                    -        */
162      119           public boolean isOpen() {
163      120               return this.periodMovementState == PeriodMovementState.OPEN;
164      121           }
165                    -
166                    -       /**
167                    -        * To check if this movement is finalized or not
168                    -        *
169                    -        * @return true if paid, false otherwise
170                    -        */
171      122           public boolean isAccounted() {
172      123               return this.periodMovementState == PeriodMovementState.ACCOUNTED;
173      124           }
```

```
174   125
175         -      /**
176         -       * To check if this movement is overdue
177         -       *
178         -       * @return true if is, false otherwise
179         -       */
180   126          public boolean isOverdue() {
181   127              return this.dueDate.isBefore(LocalDate.now());
182   128          }
183   129
184         -      /**
185         -       * Helper method to get only the discount value
186         -       *
187         -       * @return the payment discount value
188         -       */
189   130          public BigDecimal getValueWithDiscount() {
190   131              return this.payment != null ? this.payment.getPaidValue() : this.getValue();
191   132          }
```

0 comments on commit 519e6aa

Please sign in to comment.