

---

**tryton***account\_statement\_rule* Documentation  
**Release latest**

**May 06, 2019**



---

## Contents

---

<b>1</b>	<b>Rule</b>	<b>3</b>
1.1	Criteria . . . . .	3
1.2	Lines . . . . .	3



The `account_statement_rule` module allows rules to be defined to complete statement lines from imported files. When the “Apply Rule” button is clicked on a statement, each rule is tested in order against each origin that does not have any lines until one is found that matches. Then the rule found is used to create the statement lines linked to the origin.



A rule is composed of two parts: matching criteria and lines.

## 1.1 Criteria

The criteria are matched with each origin of the statement:

- Company
- Journal
- Amount: Check if the amount is between two values
- Description: A [regular expression](#) to search for a match in the origin description.
- Information rules:
  - Key: the statement information key on which the rule applies
  - The matching value depending of the type of the key:
    - \* Boolean
    - \* Numeric: A range of value.
    - \* Char: A regular expression.
    - \* Selection

The regular expression can register the group names *party*, *bank\_account* and *invoice* which are later used to search for a party and an invoice.

## 1.2 Lines

They define how to create the statement lines from the matching origin:

- **Amount: A Python expression evaluated with:**
  - *amount*: the amount of the origin.
  - *pending*: the amount from which previous lines have been deducted.
- Party
- Account

If the party is not filled in, one will be searched for using the *bank\_account* or the *party* group names from the regular expressions. If the *invoice* group name appears in a regular expression, it will be used to find an invoice to link with.