E2E: Dashboard-> Widgets Drag'n'drop

Reordering widgets on the Dashboard page

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as User (Admin, RA). Dashboard page is opened At least two widgets are created in dashboard

Step	1
Action	Click the "Gear" icon at the header of the page.
Input data	
Expected result	 All the menus are become not active. The buttons "Save", "+ New widget", "Reset to default all widgets" for admins and the "+ New widget" ca rd was displayed.
Step	2
Action	Click empty space at the top of the widget and drag it to another widget, then drop it there.
Input data	
Expected result	Widgets are swap their places.
Step	3
Action	Click the "Save" button.
Input data	
Expected result	 All the menus are become active. The buttons "Save", "+ New widget" and the "+ New widget" card, "Reset to default all widgets" for admins was disappear. The "Widget positions was successfully updated" message was displayed.

Dashboard-> Adding limit number for widgets

Adding maximum number of 10 for each widget

Severity	Priority	Behavior	Туре
✓ Minor	↓ Low	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Action

Click on the "Save" button.

There are products with competitors in the account. Open the Dashboard page. GA widgets are enabled.

Steps to reproduce		
Step	1	
Action	Click the "Gear" icon at the header of the page.	
Input data		
Expecte result	d1. All the menus are become not active. 2. The "Save", and "+ New widget" buttons, and the "+ New widget" card, (the "Reset to default all widgets" buttons on for admin) were displayed.	
Step	2	
Action	Add "Price position", "Stock status", "Selected domains", "Top domains", "Product price history", "New prices", "Inventory Value", "Competitor Activity", "Profit", "Conversion rate", "Google Analytics", "Gross margin" (admin) widgets by clicking the "+" icon to the left of a widget name (1 time for "Stock status", "New prices" and "C ompetitor activity" widgets and 10 times for other widgets).	
Input data		
Expecte result	 The count of widgets was increased to 1 or 10. The "+" icon and widget's names are grayed and unclickable. The "It`s only possible to create 10 <widget`s name=""> widget" or The "It`s only possible to create 1 <widget`s name=""> widget" tooltips are displayed.</widget`s></widget`s> 	
Step	3	
Action	Click on the "+" icon to the left of the "Price position" widget.	
Input data		
Expecte result	d. The counts of the "Price position" widget were not increasing. 2. "+" icon and widget name is grayed and unclickable.	
Step	4	

Input data

Expected 1. The "Add widget" popup was closed and the selected widgets are displayed. The "Widgets were successfully created" message was displayed. There are 93 widgets for Admin/RA, and 83 for User on the Dashboard page.

E2E: Dashboard -> "Generate default widgets" button (Admin/RA)

Adding 9 defoult widgets ("Welcome", "Price Position", "Selected Domains", "Gross margin", "Product price History", "Competitor Activity", "New Prices", "Stock Status", "Inventory Value")by clicking the "Generate defoult widgets" button

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted

Pre-conditions

Input data

- 1. Admin (RA) is logged in, switched to a non-connected account.
- 2. Dashboard page is opened, without widgets.

Step	1
Action	Generate defoult widgets.
Input data	
Expecte result	 "GENERATE DEFAULT WIDGETS" popup with "Are you sure you want to generate default widgets?" message dis played. 9 default widgets added in the following order ("Welcome", "Price Position", "Selected Domains", "Gross margin", "Product price History", "Competitor Activity", "New Prices", "Stock Status", "Inventory Value")
Step	2
Action	Delete "Price Position" widget.
Input data	
Expecte result	1. "Delete widget" popup with "Are you sure you want to delete the widget?" message displayed. 2. "Widget was successfully deleted" information message was displayed. "Price Position" widget is not displayed anymore.
Step	3
Action	Generate defoult widgets.

1. "GENERATE DEFAULT WIDGETS" popup with "Are you sure you want to generate default widgets?" message dis

Expected played.

Price Position" widget was added. Pay attention that total number of widgets is 9 ("Welcome", "Price Position", "Selected Domains", "Gross margin", "Product price History", "Competitor Activity", "New Prices", "Stock Status ","Inventory Value")

E2E: Dashboard-> "Reset to default all widgets" button (Admin/RA)

Reset all widgets to default settings.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from different domains.
- 2. Dashboard page is opened.
- 3. Google analytics is enabled, account filled with test data (connected to the real GA service). GA widgets are enabled.
- 4. "Price position", "Selected domains", "Top domains", "Product price history", "Inventory value", "Google Analytics", "Profit", "Conversion rate" widgets are added.
- 5. All widgets are filtered with custom settings.

Step	1
Action	Click the "Gear" icon at the header of the page.
Input data	
Expected result	All the menus are become not active. Widgets are displayed in editing mode (delineated red dashes lines) The buttons "Save", "+ New widget", "Reset to default all widgets" button, "Price position", "Selected domains", "Top domains", "Product price history", "Google Analytics", "Profit", "Conversion rate" widgets and the "+ New widget" card displayed
Step	2
Action	Reset to default all widgets
Input data	

- 1. "Reset to default" pop-up with "Are you sure you want to reset to default all widgets?" message displayed.
- 2. "All widgets was reset to default" information message displayed.
- 3. Pay attention to the "Price position", " "Selected domains", "Top domains", "Product price history", "Google An alytics", "Profit", "Conversion rate" widgets.

For "Price Position" widget the default condition is:

- 1) For the first pie graph Filters not installed displays "All products";
- 2) Filter by "Brand" with the largest number of products;
- 3) Filter by "Product types" with the largest number of products.

For "Selected domains" and "Top domains" widgets the default condition is displaying 15 domains in order from $Expected_{the}$ most matches without any filters. result

For "Product price history" widget the default condition is displaying 10 products with the biggest number of matches, without any filters.

For "Google Analytics" widget the default condition is displaying "Data interval" that choosed on Settings->Google Analytics->Data interval, for all products.

For "Profit" widget the default condition is displaying "Data interval" that choosed on Settings->Google Analytics->Data interval, Total profit for all products/ Top 5 products from all products/ Top 5 brands from all products.

For "Conversion rate" widget the default condition is displaying "Data interval" that choosed on Settings->Google Analytics->Data interval, Top 5 products with Highest revenue from all products.

Dashboard -> NO save request in case there are no changes in widgets

We shouldn't send a save request in case there are no changes in all widgets on the Dashboard page (except Welcome, Stock status, New prices widgets, Competitor Activity)

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Performance
Milestone	Automation	Status	Is Muted

Pre-conditions

- 1. Account has products with matches from different domains.
- 2. Dashboard page is opened.
- 3. Google analytics is enabled, account filled with test data (connected to the real GA service). GA widgets are enabled.
- 4. "Price position", " "Selected domains", "Top domains", "Product price history", "Inventory Value", "Google Analytics", "Profit", "Conversion rate" widgets are added.
- 5. DevTools Google Chrome Console is opened/ Debugger on Queries tab

Step	1
Action	Click on the red editing "Pen" icon inside any pie chart on the "Price position" widget. Click through tabs and save without changes. Pay attention to the Console/Queries tab, last requests.
Input data	
Expected result	1."CHOOSE WHAT YOU WANT TO SEE" fiter popup displayed. https://app.priceshape.dk/api/v2/facets 2."COMPETITOR STOCK FILTER" tab displayed. No new requests 3.Editing popup closed. "Widget was successfully updated" informational message displayed. 4.Dashboard page is opened, "Price position" widget is displayed without changes. No new requests displayed.
Step	2
Action	Reproduce to "Selected domains", "Top domains", steps 1,3 (witout "COMPETITOR STOCK FILTER" tab) to "Product price history", "Inventory Value", "Google An alytics", "Profit", "Conversion rate" widgets
Input data	
Expected result	NO save request in case there are no changes in widgets

Dashboard-> Filtering and search value is not passed between "New prices" and "Competitor activity" widgets

Фильтрация по поиску или по фильтрам не должна быть связана между "New prices" и "Competitor activity" виджетами.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Залогиниться как юзер (админ, РА).

В аккаунте есть продукты с матчами.

Собрана история изменения своих цен, цен и сток статуса конкурентов за последний месяц.

Ha Dashboard добавлены "New prices" и "Competitor activity" виджеты, нет сохраненных фильтров или поиска. Открыть Dashboard.

Step	1
Action	Открыть страницу "New prices" нажав "See all new prices" на "New prices" виджете.
Input data	
Expected result	"New prices" страница открыта на "Last 24 hours" вкладке.
Step	2
Action	Отфильтровать виджет по периоду "Last month" и в поле поиска ввести слово из тайтла одного из пр одуктов в списке (GTIN).
Input data	Candle
Expected result	 Виджет отображает изменения цен в период "Last month". "Candle" отображается в поле поиска. В списке отображаются изменения цен продуктов, что содержат "Candle" в тайтле.
Step	3
Action	Открыть страницу "Competitor activity" нажав "See all competitor activity" на "Competitor activity" видж ете. Обратить внимание на поле поиска.
Input data	
Expected result	1. "Competitor activity" страница открыта на "Last 24 hours", "Price" вкладке. 2. Поле поиска пустое.
Step	4
Action	Отфильтровать виджет по периоду "Last month" и в меню фильтров выбрать бренд "A".

Input data	
Expected result	Виджет отображает изменения цен конкурентов для продуктов бренда "A" в период "Last month".
Step	5
Action	Открыть страницу "New prices". Обратить внимание на поле поиска и фильтры.
Input data	
Expected result	 Виджет отображает изменения цен в период "Last month". "Candle" отображается в поле поиска. В списке отображаются изменения цен продуктов, что содержат "Candle" в тайтле. Фильтры не применены.
Step	6
Action	Открыть страницу "Competitor activity" . Обратить внимание на поле поиска и фильтры.
Input data	
Expected result	 Виджет отображает изменения цен конкурентов для продуктов бренда "А" в период "Last month". В фильтрах выбран бренд "А", другие бренды отображаются серым, задизейбленны. Поле поиска пустое.

Dashboard->Widgets->Filters->Saved filters from session to session (Admin)

Отображение "Price position", "Product price history", "Selected Domains", "Top domains", "Inventory value", "Conversion rate", "Profit", "Google Analytics" виджетов согласно фильтрам от сессии к сессии.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Залогиниться как админ/ Ра в присоединенный аккаунт.

В аккаунте есть товары с матчами.

У продуктов есть атрибуты "Brand", "Product type", "Custom label" с несколькими значениями.

Созданы и присвоены продуктам "Tag" и "Dynamic tag".

На дашборде созданы "Price position", "Product price history", "Selected Domains", "Top domains", "Inventory value", "Conversion rate", "Profit", "Google Analytics" виджеты и отфильтрованы по кастомным настройкам (комбинации"Brand", "Product type", "Custom label", "Tag" и "Dynamic tag", для "Product price history"- выбрано продукта).

Step	1
Action	Перезагрузить Dashboard страницу через F5, обратить внимание на виджеты.
Input data	
Expecte result	d ^P rice position", "Product price history", "Selected Domains", "Top domains", "Inventory value", "Conversion rate", "Profit", "Google Analytics" виджеты и отфильтрованы по кастомным настройкам (комбинации"Brand", "Prod uct type", "Custom label","Tag" и "Dynamic tag", для "Product price history"- выбрано продукта).
Step	2
Action	Выйти и зайти в систему (logout, login), зайти в нужный аккаунт, проверить виджеты на дашборде.
Input data	
Expecte result	dPrice position", "Product price history", "Selected Domains", "Top domains", "Inventory value", "Conversion rate", "Profit", "Google Analytics" виджеты и отфильтрованы по кастомным настройкам (комбинации"Brand", "Prod uct type", "Custom label","Tag" и "Dynamic tag", для "Product price history"- выбрано продукта).

Dashboard->Widgets->New Prices/Competitor activity widgets - > Filters -> "Saved filters" menu

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. После смены опции все выбранные ранее фильтра не сбрасываются.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

2

Pre-conditions

- 1. В аккаунте есть:
 - товар №1 (brand tradition, tag No tags) с новой ценой неделю назад;
 - ∘ товар №2 (brand louis, tag test) с новой ценой вчера;
 - товар №3 (brand tradition, tag test) с новой ценой сегодня;
 - о товар №4 (brand louis, tag test) с новой ценой 2 месяца назад.
- 2. Ha New Prices/Competitor activity странице есть:
 - "Test" шаблон с выбранным "tradition" фильтром по бренду;
 - "TestTag" шаблон с выбранным "test" фильтром по тегу.
- 3. Перейти на New Prices страницу.

Step	1
Action	Перейти на страницу фильтров.
Input data	
Expected result	 Открылась страница фильтров в "Brands" меню. "Saved filters" меню кликабельное.
Step	2
Action	Нажать на "Saved filters" меню.
Input data	
Expected result	1. В "Saved filters" меню отображаются "Test" и "TestTag" шаблоны.
Step	3
Action	Удалить "TestTag" шаблон.
Input data	
Expected result	1. "TestTag" шаблон удален. 2. В "Saved filters" меню отображается "Test" шаблон.
Step	4
Action	Удалить "Test" шаблон.

Input data

Expected result	 "Test" шаблон удален. В "Saved filters" меню ничего не отображается.
Step	5
Action	Перейти в "Brands" меню.
Input data	
Expected result	 Открылись фильтра в "Brands" меню. "Saved filters" меню стало некликабельным.
Step	6
Action	Повторить кейс для "Competitor activity" виджета.
Input data	
Expected result	 Шаблоны "TestTag" и "Test" удалены. В "Saved filters" меню ничего не отображается. "Saved filters" меню стало некликабельным.

Dashboard -> Widgets -> Checking the identical "profit" value in the Profit and Google analytics widget

Значения профита в "Profit" и "Google analytics" всегда должны быть одинаковыми.

Severity	Priority	Behavior	Туре
O Normal	↑ High	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	2	1	

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. В аккаунте подключена гугл аналитика, собраны данные о продуктах, у продуктов есть значения "cost price".
- 3. В настройках гугл аналитики в "Data interval "выбрано "1month".
- 4. Опция "Calculate gross profit with VAT" выключена в настройках.
 5. Добавить "Profit", "Google analytics" виджеты на Dashboard страницу.

Step	1
Action	Сравнить значение профита в виджетах "Profit" и "Google analytics".
Input data	
Expected result	Значения в виджетах идентичные.
Step	2
Action	Включить опцию "Calculate gross profit with VAT"
Input data	
Expected result	Опция включена.
Step	3
Action	Сравнить значение профита в виджетах "Profit" и "Google analytics".
Input data	
Expected result	Значения в виджетах идентичные.

E2E: Dashboard->Widgets->Option GA widget is disabled/enabled (Admin/RA)

Displaying "Profit", "Conversion rate", "Google Analytics" widgets when GA is disabled/enabled in account

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Products have filled "Cost price" attribute.
- 2. On the Datafeeds page Google analytics is enabled, data is connected or generated test data.
- 3. Dashboard page is opened.4. "Profit", "Conversion rate", "Google Analytics" widgets are added.

Step	1
Action	Pay attention to the GA widgets, click throught tabs, check data.
Input data	
Expected result	1. "Profit", "Conversion rate", "Google Analytics" widgets displays data on all tabs.
Step	2
Action	Disable integration with Google Analytics in your account
Input data	
Expected result	1. Google Analytics was disabled in account
	DATES DA
Step	3
Action	Open the Dashboard page. Pay attention to the GA widgets.
Input data	
Expected result	GA widgets are not displayed.

Dashboard -> Price position widget -> Mix Tags, Dynamic tags parameters + the missing stock status of a competitor

SeverityPriorityBehaviorTypeO NormalO MediumPositiveFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. В аккаунте есть товары с матчами.
- 2. К группе продуктов применен tag_1 и dynamic_tag_1, у них есть матчи со сток статусом In stock.
- 3. Перейти на Dashboard страницу.
- 4. Price position виджет добавлен на страницу.

Step	1
Action	Нажать на красную кнопкуредактирования внутри одной из круговых диаграмм Price position виджета.
Input data	
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". Ф. Вверху окна отображается фильтр по сток статусу конкурента. 3. Под фильтром сток статуса отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
Step	2
Action	Нажать на Tags поле и выбрать "tag_1" и "dynamic_tag_1".
Input data	
Expecte result	${ m d}_{1.}$ " ${ m tag_1}$ " и "dynamic_tag" названия отображаются в Tags поле и под Tags полем.
Step	3
Action	Активировать Out of stock.
Input data	
Expecte result	d ₁ . Выбранный чекбокс стал активным.
Step	4
Action	Нажать на Next кнопку.

Input data

1. Вверху окна отображается фильтр по сток статусу конкурента.

result

- Expected 3. Отображается фильтр доменов. 3. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do mains, all domains are selected and new domains will be auto selected" message".
 - 4. Кнопки "Resert to default", "Back" и "Save"активны.

Step 5

Action Выбрать в дропдауне любой домен.

Input data

lampekartellet.dk

Expected. Чекбокс домена стал активным.

result 2. Текст внизу изменился на "Selected 1 domain(s)".

6 Step

Action Сохранить изменения.

Input data

Expected 1. У выбранного параметра активирован чекбокс. 1. Появилось информационное сообщение «Widget updated successfully" is displayed». result 3. Отображается сообщение "No data", диаграмма - кольцо не отображается.

Dashboard -> Price position widget -> Display applied filters in hover

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

2

Pre-conditions

- 1. У продукта №1 есть бренд "А", продукт тип "В", Тег "one", DT "DT_1", Label "Custom",также у продукта есть матч с домена "test.dk со статусом "в наличии". 2. Страница "Dashboard"открыта.
- 3. Добавлен виджет "Price position".

Steps to reproduce			
Step	1		
Action	Нажать красную кнопку «Редактировать» внутри одной из круговых диаграмм виджета "Price position".		
Input data			
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". Ф. Вверху окна отображается фильтр по сток статусу конкурента. 3. Под фильтром сток статуса отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.		
Step	2		
Action	Нажать на поле Теги» и выбрать тег " one" и динамический тег "DT_1"		
Input data			
Expecte result	dИмена " tag_1" и "dynamic_tag" отображается в поле поиска «Tags», а также под поиском с красным кресто м слева.		
Step	3		
Action	Нажать на поле «Brands» и выбрать бренд "А"		
Input data			
Expecte result	d 1. Имя бренда "А" отображается в поле поиска «Brands», а также под поиском с красным крестом слева.		
Step	4		
Action	Нажать на поле «Product Types» и выбрать продукт тип "В"		

Input data		
Expected result		мя типа "В" отображаются в поле поиска «Product Types», а также под поиском с красным крестом сле a.
Step	5	
Action	Наж	кать на поле «Labels» и выбрать лейбл "Custom"
Input data		
Expected result	d1. И ва	мя лейбла "Custom" отображается в поле поиска «Labels», а также под поиском с красным крестом сле a.
Step	6	
Action	Наж	кать "Next" кнопку
Input data		
Expected result	d2. O 3. B m	верху окна отображается фильтр по сток статусу конкурента. тображается фильтр доменов. нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do nains, all domains are selected and new domains will be auto selected" message". нопки "Resert to default", "Back" и "Save"активны.
Step	7	
Action		ыбрать "In stock" статус ыбрать домен "test.dk".
Input data		
Expected result	о <mark>2. Ч</mark> 3. Д 4. "F	параметра сток статуса активирован чекбокс. екбокс домена также активирован. омен "test.dk" отображается внизу поля поиска доменов с красным крестом. Please be aware that if you don't select any domains, all domains are selected and new domains will be auto s lected" сообщение сменилось на счетчик "Selected 1 domain(s)"
Step	8	
Action	Cox	ранить изменения.
Input data		
Expected result		оявилось информационное сообщение «Widget updated successfully" is displayed». тображается внутри диаграммы "1 products"
Step	9	
Action	Нав	ести курсором на названия примененных фильтров внизу диаграммы

Отображается тултип, в котором указано, что выбрано:

- Сток статус: "In stock" и "Out of stock"
 Бренд "А"

 Expected Продукт тип "В"
 result Ter " one"

- Динамический тег "DT_1"Лейбл "Custom"

Выбранный домен "test.dk" НЕ отображается в тултипе.

Dashboard -> Price position widget -> General agenda

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами.
- 2. В настройках установлены такие параметры (Account setup)
 - Lowest <22%
 - Semi low 22%-3%
 - Neutral 3%-1%
 - Semi hight 1%-25%
 - Highest >25%
- 3. Страница "Dashboard"открыта.
- 4. Виджет "Price position" добавлен.

Step	1
Action	Навести курсором на "Lowest" позицию вверху диаграммы
Input data	
Expected result	На всех трех диаграммах одновременно подсветились "Lowest" секции с процентным соотношением пр одуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	2
Action	Навести курсором на "Semi low" позицию вверху диаграммы
Input data	
Expected result	На всех трех диаграммах одновременно подсветились "Semi low" секции с процентным соотношением продуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	3
Action	Навести курсором на "Neutral" позицию вверху диаграммы
Input data	
Expected result	Ha всех трех диаграммах одновременно подсветились "Neutral секции с процентным соотношением пр одуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	4
Action	Навести курсором на "Semi hight" позицию вверху диаграммы

Input data

Expected result	Ha всех трех диаграммах одновременно подсветились "Semi hight" секции с процентным соотношение м продуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	5
Action	Навести курсором на "Highest" позицию вверху диаграммы
Input data	
Expected result	Ha всех трех диаграммах одновременно подсветились "Highest" секции с процентным соотношением п родуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"

Dashboard -> Price position widget -> Redirecting to the Product list page

Redirecting to the Product list page with selected price position filters on the "Price position" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с совпадениями и разными ценовыми позициями.
- 2. В настройках установлены такие параметры (Account setup)
 - Lowest <22% (7 products)
 - Semi low 22%-3% (5 products)
 - Neutral 3%-1% (79 products)
 - Semi hight 1%-25% (179 products)
 - Highest >25% (146 products
- 3. Страница "Dashboard" открыта.
- 4. Виджет "Price position" добавлен, в нем отображается 5 секций.

Step	1		
Action	Нажать на "Lowest " секцию диаграммы "All products"		
Input data			
Expected result	 Количество продуктов на секции "Lowest " диаграммы "All products" совпадает со значениями в "Product list". Значения "diff", которые отображаются на карточках конкурентов, соответствуют диапазону "Low" <22%. 		
Step	2		
Action	Нажать на "Semi low" секцию диаграммы "All products"		
Input data			
Expected result	 Количество продуктов на секции "Semi low" диаграммы "All products" совпадает со значениями в "Pr oduct list". Значения "diff", которые отображаются на карточках конкурентов, соответствуют диапазону "Semi low" 22%-3%. 		
Step	3		
Action	Нажать на "Neutral " секцию диаграммы "All products"		
Input data			
Expected result	 Количество продуктов на секции "Neutral " диаграммы "All products" совпадает со значениями в "Pr oduct list". Значения "diff", которые отображаются на карточках конкурентов, соответствуют диапазону "Neutr al" 3%-1%. 		

Step	4
Action	Нажать на "Semi hight " секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Semi hight" диаграммы "All products" совпадает со значениями в " Product list". Значения "diff", которые отображаются на карточках конкурентов, соответствуют диапазону "Semi hight" 1%-25%.
Step	5
Action	Нажать на "Highest" секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Highest " диаграммы "All products" совпадает со значениями в "Pr oduct list". Значения "diff", которые отображаются на карточках конкурентов, соответствуют диапазону "Highe st" >25%.

Dashboard -> Price position widget -> Changing Price position parameters on the Settings page

Check correct displaying parameters on the "Price position" widget and redirecting to the Product list page after changing parameters on the Settings page

Severity	Priority	Behavior	Туре
^ Major	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted

Pre-conditions

- 1. Log in as a User. There are products with matches, different brands, tags, product types, labels and different price positions in the account.
- 2. On the Setting page "Account setup" next price positions were set (example):

Lowest <22% Semi low 22%-3% Neutral 3%-1% Semi hight 1%-25% Highest >25%

3. "Price position" widget was added. Dashboard page is opened.

1
Hover a cursor over a different sectors on the first pie chart with "All Products". Check correct display of price p ositions and amount of the products. Click on the red "Highest price" sector of the first pie chart
dl. The number of products is the same as in the pre-conditions. Redirection to the Product list page, where 146 pr oducts are displayed, all the first matches cards have diff > -25 with our product
2
Open the Settings page
${ m d}_{ m 1.}$ Settings page is opened
3
Click on the "Account setup" menu
${ m d}_{ m l}$. The field with menu is opened. Price position "thermometer" settings are displayed

Step	4
Action	Change current price position settings to. Click on "Save changes" button
Input data	Lowest <10% Semi low 10%-1% Neutral 1%-5% Semi hight 5%-50% Highest >50%
Expected result	1. New setup on the Price position "thermometer" is displayed. "Settings saved" information message was display ed
Step	5
Action	Open the Product List page.
Input data	
Expected result	1. Product List page is opened.
Step	6
Action	Filter products by "Highest price" price position
Input data	
Expected result	1. Products with "Highest price" position with diff > -50% are displayed. Diff with percantage are marked in red col or. The amount of products is the same as on Filters pop-up and on Settings page for "Highest price" price posit ion
Step	7
Action	Filter products by "Semi hight price" price position
Input data	
Expected result	1. Products with "Semi hight price" position with diff from -5 to -50% are displayed. Diff with percantage are marked in bright-red color. The amount of products is the same as on Filters pop-up and on Settings page for "Semi hight price" price position
Step	8
Action	Filter products by "Neutral price" price position
Input data	
Expected result	1. Products with "Neutral price" position with diff from 1% to -5% are displayed. Diff with percantage are marked in grey color. The amount of products is the same as on Filters pop-up and on Settings page for "Grey price" price position
Ston	

Step 9

Action	Filter products by "Semi low price" price position
Input data	
Expected result	1. Products with "Semi low price" position with diff from 1 to -10% are displayed. Diff with percantage are marked in bright-green color. The amount of products is the same as on Filters pop-up and on Settings page for "Semi low price" price position
Step	10
Action	Filter products by "Lowest price" price position
Input data	
Expected result	 Products with "Lowest price" position with diff <10% are displayed. Diff with percantage are marked in green co lor. The amount of products is the same as on Filters pop-up and on Settings page for "Lowest price" price posit ion

Dashboard->"Price position" widget->Check data in widget

Данны тест кейс проверяет верность данных в виджете Price postion

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами
- 2. Свои продукты имеют Brand
- 3. Открыта страница Dashboard {Host}/
- 4. Добавленный виджет Price Positions с дефолтными фильтрами
- 5. В настройках аккаунта выставленные дефолтные настройки к градуснику Price position LOWEST PRICE = 0

NEUTRAL PRICE between 0 and 0 HIGHEST PRICE 0

Step	1
Action	Проверить результат первой диаграмы по забросу
Input data	SELECT CASE WHEN matchings_data ->> '\$."min_diff"' < 0 THEN 'LOWEST PRICE' WHEN matchings_data ->> '\$."min_diff"' = 0 THEN 'NEUTRAL PRICE' ELSE 'HIGHEST PRICE' END AS min_diff_group, COUNT(*) AS count FROM sphinx_products WHERE account_id =account_id AND matchings_data IS NOT null GROUP BY min_diff_group
Expected result	1. Данные с базы совпадаю с данными в виджете по периодам
Step	2
Action	В настройках аккаунта изменить данные в градуснику Price position
Input data	LOWEST PRICE = 5 SEMI LOW PRICE between 5 and 1 NEUTRAL PRICE between 1 and 0 SEMI HIGH PRICE between 0 and 10 HIGHEST PRICE 10
Expected result	1. Выбранные данны успешно сохраннены в настройках аккаунта
Step	3
Action	Открыть страницу Dashboard и сравнить результаты в виджете с базой данных

SELECT CASE WHEN matchings_data ->> '\$."min_diff"' > 5 THEN 'LOWEST PRICE'
WHEN matchings_data ->> '\$."min_diff"' <= 5 AND matchings_data ->> '\$."min_diff"' > 1 THEN 'SEMI LOW PRICE' WHEN matchings data ->> '\$."min diff"' >= 0 AND matchings data ->> '\$."min diff"' <= 1 THEN 'NEUTRAL PRICE' WHEN matchings_data ->> '\$."min_diff"' >= -10 AND matchings_data ->> '\$."min_diff"' < 0 THEN 'SEMI HIGH PRICE' Input data ELSE 'HIGHEST PRICE' END AS min diff group, COUNT(*) AS count FROM sphinx_products WHERE account_id =account_id AND matchings_data IS NOT null GROUP BY min_diff_group; 1. Данные с базы совпадаю с данными в виджете по периодам Expected result 4

Step

Action Сравнить вданные в во второй диаграме виджета

> SELECT CASE

WHEN matchings_data ->> '\$."min_diff"' > 5 THEN 'LOWEST PRICE'

WHEN matchings_data ->> '\$."min_diff"' <= 5 AND matchings_data ->> '\$."min_diff"' > 1 THEN 'SEMI LOW PRICE'

WHEN matchings_data ->> '\$."min_diff"' >= 0 AND matchings_data ->> '\$."min_diff"' <= 1 THEN 'NEUTRAL PRICE'

WHEN matchings_data ->> '\$."min_diff"' >= -10 AND matchings_data ->> '\$."min_diff"' < 0 THEN 'SEMI HIGH PRICE'

Input data ELSE 'HIGHEST PRICE'

END AS min_diff_group, COUNT(*) AS count FROM sphinx_products

WHERE account_id =account_id and brand_id =brand_id AND matchings_data IS NOT null

GROUP BY min_diff_group;

1. Данные с базы совпадаю с данными в виджете по периодам Expected result

E2E: Dashboard -> Price position widget -> Brands parameters

Filtering the Price position widget by Brands parameters

Severity	Priority	Behavior	Туре
^ Major	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches, different brands, and different price positions in the account.
- The Dashboard page is opened.
 The default Price position widget is added (the number of products is equal to the total number of products with matches in the account).

Steps to reproduce		
Step	1	
Action	Click on the Edit button inside one of the diagrams on the Price position widget.	
Input data		
Expected result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your competitor's stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Reset to default" and "Next" are active. 	
Step	2	
Action	Click on the brand's field and select one of the brands (or write the brand name in the search field and select i t by checking).	
Input data	&tradition	
Expected result	 A list of product brands with matches is displayed. The brand name is displayed in the Brands search field, as well as under the Brands search field with a cross on the left. 	
Step	3	
Action	Activate In stock or Out of stock checkbox.	
Input data	In stock	
Expected result	1. A chosen checkbox is active.	
Step	4	
Action	Click on the Next button.	
Input data		

2. Domains filter is displayed. Expected 3. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selecte result d and new domains will be auto selected" message. 4. The buttons "Resert to default", "Back" and "Save" are active. 5 Step Action Activate any domain checkbox. Input data lampekartellet.dk Expected 1. A chosen checkbox is active. 2. The text at the bottom has changed to "Selected 1 domain(s)". result 6 Step Click on the Save button. Action Input data 1. The informational "Widget updated successfully" message is displayed. Expected 2. The diagram displays the Brand's name, the number of goods in accordance with their price position, and com result petitors' stock status 7 Step Hover over the selected options below the diagram. Action Input data 1. The tooltip that appears displays: Expected - Brand: &tradition result - Stock status: In Stock Step 8 Action Follow the hyperlink on the Price position widget to Product list page. Input data Expected $_{\mbox{\scriptsize ...}}$ 1. Filtered products by price position and domain are displayed. result Step Open the Product List page, and select the same brand, price position, domain, and competitor status in the fil Action Input data Expected $_{\rm ii}$ 1. Filtering results from Price position widget and Product list are the same. result

1. Below the name of the popup, you can see a filter by your competitor stock status.

E2E: Dashboard -> Price position widget -> Product Types parameters

Setting "Price position" widget by "Product Types" parameters

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches, different product types, and different price positions in the account.
- The Dashboard page is opened.
 The "Price position" widget is added.

	Steps to reproduce		
Step	1		
Action	Click on the red pen "Edit" button inside one of the diagrams on Price position widget		
Input data			
Expected result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your competitor stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active. 		
Step	2		
Action	Click on the product type`s field and select one of the product types (or write the product type name in the se arch field and select it by checking).		
Input data	Pendler > Louis Poulsen		
Expected result	1. The product type name is displayed in the Product Type search field, as well as under the search field with a cross on the left.		
Step	3		
Action	Activate In stock or Out of stock checkbox.		
Input data	a Out of stock		
Expected result	1. A chosen checkbox is active.		
Step	4		
Action	Click on the Next button.		

Expected 2. Domains filter is displayed. 3. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selecte result d and new domains will be auto selected" message. 4. The buttons "Resert to default", "Back" and "Save" are active. 5 Step Action Activate any domain checkbox. Input data lampekartellet.dk Expected 1. A chosen checkbox is active. 2. The text at the bottom has changed to "Selected 1 domain(s)". result 6 Step Click on the Save button. Action Input data ${\bf 1.} \ \ {\bf The informational "Widget updated successfully" message is displayed.}$ Expected 2. The diagram displays the Product type's name, the number of goods in accordance with their price position, a result nd competitors' stock status. 7 Step Hover over the selected options below the diagram. Action Input data 1. The tooltip that appears displays: Expected - Product Types: Pendler > Louis Poulsen result - Stock status: Out Of Stock Step 8 Action Follow the hyperlink on the Price position widget to Product list page. Input data Expected $_{\mbox{\scriptsize ...}}$ 1. Filtered products by price position and domain are displayed. result Step Open the next tab Product List page, and select the same product type, price position, domain and competitor Action status in the filters. Input data Expected 1. Filtering results from Price position widget and Product list are the same. result

1. Below the name of the popup, you can see a filter by your competitor stock status.

E2E: Dashboard -> Price position widget -> Tags parameters

Setting "Price position" widget by "Tags" parameters

Behavior Severity **Priority Type** O Normal O Medium Positive **Functional** Milestone **Automation Status** Is Muted Manual Actual No

Pre-conditions

- 1. There are products with matches, different product types and different price positions in the account.
- The Dashboard page is opened.
 The "Price position" widget is added.

Step	1		
Action	Click on the red pen "Edit" button inside one of the diagrams on Price position widget		
Input data			
Expected result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your competitor stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active. 		
Step	2		
Action	Click on the tag`s field and select one of the tags (or write the tag name in the search field and select it by ch ecking).		
Input data	test		
Expected result	1. The tag name is displayed in the Tags search field, as well as under the search field with a cross on the left.		
Step	3		
Action	Activate In stock or Out of stock checkbox.		
nput data	Out of stock		
Expected result	1. A chosen checkbox is active.		
Step	4		
Action	Click on the Next button.		
nput data			

2. Domains filter is displayed. Expected 3. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selecte result d and new domains will be auto selected" message. 4. The buttons "Resert to default", "Back" and "Save" are active. 5 Step Action Activate any domain checkbox. Input data lampekartellet.dk Expected 1. A chosen checkbox is active. 2. The text at the bottom has changed to "Selected 1 domain(s)". result 6 Step Click on the Save button. Action Input data ${\bf 1.} \ \ {\bf The informational "Widget updated successfully" message is displayed.}$ Expected 2. The diagram displays the tag's name, the number of goods in accordance with their price position, and comp result etitors' stock status. 7 Step Hover over the selected options below the diagram. Action Input data 1. The tooltip that appears displays: Expected - Tags: test result - Stock status: Out Of Stock Step 8 Action Follow the hyperlink on the Price position widget to Product list page. Input data Expected $_{\mbox{\scriptsize ...}}$ 1. Filtered products by price position and domain are displayed. result Step Open the next tab Product List page, and select the same tag, price position, domain and competitor status in Action the filters. Input data Expected $_{\rm ii}$ 1. Filtering results from Price position widget and Product list are the same. result

1. Below the name of the popup, you can see a filter by your competitor stock status.

E2E: Dashboard -> Price position widget -> Dynamic tags parameters

Severity **Priority Behavior** Type O Normal O Medium Positive **Functional** Milestone **Automation** Is Muted **Status** Manual Actual

Pre-conditions

- There are products with matches, different product types and different price positions in the account.
 The Dashboard page is opened.
- 3. The "Price position" widget is added.

Cton	1		
Step	1		
Action	Click on the red pen "Edit" button inside one of the diagrams on the Price position widget		
Input data			
Expected result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your competitor stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active. 		
Step	2		
Action	Click on the dynamic tag`s field and select one of the dynamic tags (or write the dynamic tag`s name in the s earch field and select it by checking).		
Input data	a Test		
Expected result	1. The dynamic tag name is displayed in the Dynamic Tags search field, as well as under the search field with a cross on the left.		
Step	3		
Action	Activate In stock or Out of stock checkbox.		
Input data	a In stock		
Expected result	1. A chosen checkbox is active.		
Step	4		
Action	Click on the Next button.		
Input data			

Expected 2. Domains filter is displayed. 3. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selecte result d and new domains will be auto selected" message. 4. The buttons "Resert to default", "Back" and "Save" are active. Step 5 Action Activate any domain checkbox. Input data lampekartellet.dk Expected 1. A chosen checkbox is active. 2. The text at the bottom has changed to "Selected 1 domain(s)". result 6 Step Click on the Save button. Action Input data The informational "Widget updated successfully" message is displayed.
 The diagram displays the dynamic tag`s name, the number of goods in accordance with their price position, a Expected result nd competitors' stock status. 7 Step Action Follow the hyperlink on the Price position widget to Product list page. Input data Expected $_{\cdot\cdot\cdot}$ 1. Filtered products by price position and domain are displayed. result 8 Step Open the next tab Product List page, and select the same dynamic tag, price position, domain and competitor

1. Below the name of the popup, you can see a filter by your competitor stock status.

Input data

status in the filters.

Action

result

Expected 1. Filtering results from Price position widget and Product list are the same.

E2E: Dashboard -> Price position widget -> Mix Tags and Dynamic tags parameters

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами.
- 2. К группе продуктов применен tag_1 и dynamic_tag_1, у них есть матчи со сток статусами In stock, Out of stock.
- 3. Dashboard страница открыта.
- 4. Price position виджет добавлен на страницу.

Step	1
эсер	-
Action	Нажать на красную кнопкуредактирования внутри одной из круговых диаграмм Price position виджета.
Input data	
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". cd. Вверху окна отображается фильтр по сток статусу конкурента. 3. Под фильтром сток статуса отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
Step	2
Action	Нажать на Tags поле и выбрать "tag_1" и "dynamic_tag_1".
Input data	tag - tag_1 dynamic tag - dynamic_tag_1
Expecte result	ed. " tag_1" и "dynamic_tag" названия отображаются в Tags поле и под Tags полем.
Step	3
Action	Активировать любой чекбокс сток статусов конкурентов.
Input data	In stock
Expecte result	d. Выбранный чекбокс стал активным.
Step	4

Input data	
Expecte result	 Вверху окна отображается фильтр по сток статусу конкурента. Отображается фильтр доменов. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do mains, all domains are selected and new domains will be auto selected" message". Кнопки "Resert to default", "Back" и "Save"активны.
Step	5
Action	Выбрать в дропдауне любой домен.
Input data	lampekartellet.dk
Expecte result	dl. Чекбокс домена стал активным. 2. Текст внизу изменился на "Selected 1 domain(s)".
Step	6
Action	Сохранить изменения.
Input data	
Expecte result	1. Отображается "Widget updated successfully" сообщение. 2. На диаграмме отображаются названия тега и динамического тега, количество товаров в соответствии с их ценовой позицией, а также сток статуса конкурентов.
Step	7
Action	Наведите курсором на выбранные параметры под диаграммой.
Input data	
Expecte result	 1. В появивашемся тултипе отображается: d - Tags: &tag_1 - Dynamic tags: dynamic_tag_1 - Stock status: In Stock
Step	8

Перейти по гиперссылке Price position виджета на Product list страницу.

 $\mathsf{Expected}_{\mathsf{L}}$. Отображаются отфильтрованные товары по ценовой позиции, $\mathsf{tag_1}$ тегу и $\mathsf{dynamic_tag_1}$ динамическом

В соседней вкладке открыть Product list страницу, выбрать в фильтрах динамический тег - dynamic_tag_

у тегу, а также по выбранному сток статусу конкурента выбранного домена.

1, тег - tag_1, ценовую позицию, сток статус конкурента, выбранный домен.

Action

Input data

result

Step

Action

9

Input data

Expected Peзультаты перехода по гиперссылке из Price position виджета и на Product list странице совпадают. result

E2E: Dashboard -> Price position widget -> Imported tags parameters

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported_tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре	
O Normal	O Medium	Positive	Functional	
Milestone	Automation	Status	Is Muted	
-	Manual	Actual	No	
1	1	1		

Pre-conditions

- 1. В аккаунте есть продукт №1 и №2 с матчами.
- 2. У продукта №1 импортные теги test1, test2. У продукта №2 импортные теги test3, test4.
- 3. На странице "Dashboard" добавлен "Price position" виджет.
- 4. Открыть страницу "Dashboard".

Step	1
Action	Открыть режим редактирования одной из диаграм "Price position" виджета.
Input data	
Expecte result	d _{Открылся} "EDIT PRICE POSITION WIDGET" попап.
Step	2
Action	Раскрыть фильтр "Tag" -> "Imported tags"
Input data	
Expecte result	^d Отображаются импортные теги всех продуктов с матчами в аккаунте.
Step	3
Action	Выбрать импортный тег "test1"
Input data	
Expecte result	d Импортный тег "test1" отображается под фильтром "Tag"

Step	4
Action	Нажать "Next" кнопку.
Input data	
Expecte result	ed Отображается фильтр по доменам.
Step	5
Action	Нажать "Save" кнопку.
Input data	
Expecte result	edl. Отображается "Widget was successfully updated" сообщение. 2. В диаграмме отображается "1 products"
Step	6
Action	Выполнить запрос в БД.
Input data	select count(*) as Products from sphinx_products inner join product_imported_tags on sphinx_products.id = pro duct_imported_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id where imported_tags.name = ' imported_tags_name ' and sphinx_products.matchings_data is not null and sphinx_products.account_id = account_id ;
Expecte result	ed Результат совпадает.

E2E: Dashboard -> Price position widget -> "Labels" parameters

Setting "Price position" widget by "Labels" parameters

Behavior Severity **Priority Type** O Normal O Medium Not set Other Is Muted Milestone **Automation Status** Manual Actual No

Pre-conditions

- 1. There are products with matches, different labels (Custom labels) and different price positions in the account.
- Dashboard page is opened.
 "Price position" widget is added.

red pen "Edit" button inside one of the pie charts on "Price position" widget what you want to see and fill out at least one category" popup is displayed. name of the popup, you can see a filter by your competitor stock status. status stock filter, filters by product attributes are displayed. ns "Resert to default" and "Next" are active.
name of the popup, you can see a filter by your competitor stock status. status stock filter, filters by product attributes are displayed. ns "Resert to default" and "Next" are active.
name of the popup, you can see a filter by your competitor stock status. status stock filter, filters by product attributes are displayed. ns "Resert to default" and "Next" are active.
"Labels" field and select one of the labels (or write the label name in the search field and select it
"Labels" field and select one of the labels (or write the label name in the search field and select it
ame is displayed in the "Labels" search field, as well as under the search field with a cross on the lef
tock" (or "Out of stock") parameter
appears next to the selected option
'Next" button

Step 5 Action Activate any domain checkbox. Input data lampekartellet.dk Expected 1. A chosen checkbox is active. 2. The text at the bottom has changed to "Selected 1 domain(s)". result 6 Step Click on the "Save" button Action Input data The informational message "Widget updated successfully" is displayed. Notice the edited pie chart.
 It displays the Label's name, quantity of goods in accordance with their price position and competitors stock s Expected result tatus 7 Step Action Follow the hyperlink on the Price position widget to Product list page. Input data Expected $_{\cdot\cdot\cdot}$ 1. Filtered products by price position and domain are displayed. result 8 Step Open the next tab Product List page, and select the same label, price position, domain and competitor status i Action n the filters. Input data Expected 1. Filtering results from Price position widget and Product list are the same.

1. Below the name of the popup, you can see a filter by your competitor stock status.

d and new domains will be auto selected" message. 4. The buttons "Resert to default", "Back" and "Save" are active.

3. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selecte

Expected 2. Domains filter is displayed.

result

result

E2E: Dashboard -> Price position widget -> "Competitor Stock Filter" parameters

Filtering "Price Position widget" pie charts according to "In stock" and "Out of stock" statuses

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть 3 товара.
- 2. У товара №1 самый дешевый конкурент in_stock.
- 3. У товара №2 самый дешевый конкурент out_of_stock 4. У товара №3 самый дешевый конкурент без сток статуса.
- 5. Dashboard страница открыта.
- 6. Price position виджет добавлен на страницу.

Step	1
Action	Отфильтровать первую секцию по in_stock фильтру.
Input data	
Expected result	1. Внутри диаграммы отображается "2 products"
Step	2
Action	Отфильтровать вторую секцию по out_of_stock фильтру.
Input data	
Expected result	1. Внутри диаграммы отображается "1 products"
Step	3
Action	Отфильтровать третью секцию по out_of_stock и in_stock фильтру.
Input data	
Expected result	1. Внутри диаграммы отображается "3 products"

E2E: Dashboard -> Price position widget -> Several parameters

Setting "Price position" widget by several parameters

Behavior Severity **Priority Type** O Normal O Medium Not set Regression Milestone **Automation Status** Is Muted Manual Actual No

Pre-conditions

- 1. Log in as a User. There are products with matches, different brands, tags, product types, labels (custom labels) and different price positions in the account.
- - -"B" product type
 - -"C" tag
 - -"D" custom label
 - -lampekartellet.dk match with "out of stock" status .
- 3. Dashboard page is opened.

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Price position" widget
Input data	1
Expected result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your competitor stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Brands" field and select "A" brand (or write the brand name "A" in the search field and select it by check)
Input data	1
Expected result	The brand's name "A" is displayed in the "Brands" search field , as well as under the search field with a cross on the left.
Step	3
Action	Click on the "Product types" field and select "B" product type (or write the product type name in the search fi eld and select it by check)
Input data	
Expected result	The product type "B"name is displayed in the "Product types" search field, as well as under the search field with a cross on the left.
Step	4

Action	Click on the "Tags" field and select "C" tag (or write the tag name in the search field and select it by check)
Input data	
Expected result	The tag's name "C" is displayed in the "Tags" search field, as well as under the search field with a cross on the le ft.
Step	5
Action	Click on the "Labels" field and select "D" label (or write the label name in the search field and select it by check)
Input data	
Expected result	The label's name "D" is displayed in the "Labels" search field, as well as under the search field with a cross on the left.
Step	6
Action	Click on the "Next" button
Input data	
Expected result	 Below the name of the popup, you can see a filter by your competitor stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message. The buttons "Resert to default", "Back" and "Save" are active.
Step	7
Action	Activate any domain checkbox.
Input data	lampekartellet.dk
Expected result	 A chosen checkbox is active. The text at the bottom has changed to "Selected 1 domain(s)".
Step	8
Action	Choose "Out of stock" parameter
Input data	
Expected result	A checkmark appears next to the selected option
Step	9
Action	Click on the "Save" button
Input data	
Expected result	The informational message "Widget updated successfully" is displayed. In pie chart It displays selected parameters, quantity of goods in accordance with their price position and competitors stock status.

Step	10
Action	Click on the highest price position segment parameter
Input data	
Expected result	Product list with selected parameters is open. The "A" product is displayed

E2E: Dashboard -> Price position widget -> Removed filters

Severity **Priority Behavior Type Functional** O Normal O Medium **Positive** Milestone **Automation** Status Is Muted November 1.0 (2022) Manual Actual No

Pre-conditions

- 1. В аккаунте есть продукт №1 с примененный "Test" тегом и продукт №2 с примененным "DT test" динамическим тегом.
- 2. В "Price position" виджете в первой диаграмме выбрано в фильтрах "Test" тег, во второй диаграмме "DT test" динамических тег.
- 3. Перейти на Dashboard страницу.

Steps to reproduce

Step	1
Action	В "Price position" виджете навести курсором на название первой диаграммы внизу.
Input dat	a
Expected result	 В названии отображается: "Test". При наведении курсором в тултипе отображается: "Tags: Test". В диаграмме отображается: 1 products.
Step	2
Action	В "Price position" виджете навести курсором на название второй диаграммы внизу.
Input dat	a
Expected result	 В названии отображается: "DT test". При наведении курсором в тултипе отображается: "Dynamic Tags: DT test". В диаграмме отображается: 1 products.
Step	3
Action	Ha Product list странице активировать чекбокс продукта №1 -> "Select action" -> "Tags" -> "Add tag".
Input dat	a
Expected result	1. Ha Product list странице открылся "Add tag" попап, где отображается "Test" тег.
Step	4
Action	Нажать на значок корзины справа от "Test" тега и нажать на "Yes" кнопку.
Input dat	a
Expected result	Открылся "Delete tag" попап. После удаления тега "Delete tag" попап закрылся. Отображдатся "Tag deleted" сообщения.

3. Отображается "Tag deleted" сообщение.

Step	5
Action	Ha Dynamic tags странице удалить "DT test" динамический тег.
Input data	9
Expected result	1. Отображается "Dynamic tag "Test" deleted" сообщение.
Step	6
Action	Перейти на Dashboard страницу и навести курсором на название первой и второй диаграмм в "Price po sition" виджете.
Input data	a
Expected result	 В "Price position" виджете в первой и второй диаграммах отображается: No data В названии первой диаграммы отображается: "Test (removed)", при наведении на название в тултипе отображается: "Tags: Test (removed)". В названии второй диаграммы отображается: "DT test (removed)", при наведении на название в тултиге отображается: "Dynamic tags: DT test (removed)".
Step	7
Action	Нажать на кнопку редактирования первой диаграммы и нажать на "Reset to default" кнопку.
Input data	a
Expected result	1. В диаграмме отображается кол-во всех продуктов в аккаунте, в названии диаграммы отображается: " All products".
Step	8
Action	Нажать на кнопку редактирования второй диаграммы и нажать на "Reset to default" кнопку.
Input data	3

Expected 1. В диаграмме отображается кол-во всех продуктов в аккаунте, в названии диаграммы отображается: " result All products".

E2E: Dashboard -> Price position widget -> "Reset to default" button

Reset to default settings "Price position" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches and different price positions in the account.
- 2. "Price position" widget is added and filtred by custom settings (Brands, Product types, Tags, Labels, stock status and domains was choosen on 3 pie charts).
- 3. Dashboard page is opened.

```
Steps to reproduce
Step
          1
          Check result on database
Action
          SELECT * FROM `widgets` WHERE `account id` = <span style="color: var(--md-font-color-failed)">account id<
Input
          /span> and user id = <span style="color: var(--md-font-color-failed)">user id</span> AND id = id your groos
data
          s margin widget
          "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
          "brandsIds": [
          <span style="color:var(--md-font-color-failed)"> brand_id</span>
          <span style="color:var(--md-font-color-failed)"> label_id</span>
          "domainsIds": [
          <span style="color:var(--md-font-color-failed)"> domain_id</span>
          <span style="color:var(--md-font-color-failed)"> dynamic_tag_id</span>
          ],
          "mainFilters": null,
          "customLabels": [
          <span style="color:var(--md-font-color-failed)"> "name"</span>
          "customFilters": null,
          "productTypesIds": [
          <span style="color:var(--md-font-color-failed)">product type id</span>
          "competitorsStockStatus": [
          <span style="color:var(--md-font-color-failed)">"competitors Stock Status"/span>
          ]
          },
          "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
          <span style="color:var(--md-font-color-failed)"> brand_id</span>
          "labelsIds": [
          <span style="color:var(--md-font-color-failed)">label_id</span>
          "domainsIds": [
          <span style="color:var(--md-font-color-failed)"> domain id</span>
          "dynamicTags": [
```

```
Expected<span style="color:var(--md-font-color-failed)">dynamic_tag_id</span>
result
           "mainFilters": null,
           "customLabels": [
           <span style="color:var(--md-font-color-failed)">"name"</span>
           "customFilters": null,
           "productTypesIds": [
           <span style="color:var(--md-font-color-failed)">product_type_id</span>
           "competitorsStockStatus": [
           "<span style="color:var(--md-font-color-failed)">competitors_Stock_Status"</span>
          1
          },
           "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
           "brandslds": [
           <span style="color:var(--md-font-color-failed)"> brand_id</span>
           "labelsIds": [
           <span style="color:var(--md-font-color-failed)"> label id</span>
           "domainsIds": [
           <span style="color:var(--md-font-color-failed)">domain_id</span>
           "dynamicTags": [
           <span style="color:var(--md-font-color-failed)">dynamic_tag_id</span>
          ],
           "mainFilters": null,
           "customLabels": [
           <span style="color:var(--md-font-color-failed)">"name"</span>
           "customFilters": null,
           "productTypesIds": [
           <span style="color:var(--md-font-color-failed)"> product_type_id</span>
           "competitorsStockStatus": [
           <span style="color:var(--md-font-color-failed)"> "competitors_Stock_Status"</span>
          ]
           }
          ]
Step
          2
          1. Click on the red pen "Edit" button inside the first pie chart on "Price position" widget
Action
          2. Click on "Reset to default" button
Input
data
Expected. "Widget was succession, and the Price post."

The widget editing popup is closed.
             "Widget was successfully updated" information message was displayed
          3. First pie chart It shows the Price positions for all products
          3
Step
          1. Click on the red pen "Edit" button inside the second pie chart on "Price position" widget
Action
          2. Click on "Reset to default" button
Input
data
Expected. "Widget was succession updated. The widget editing popup is closed.
             "Widget was successfully updated" information message was displayed
          3. Second pie chart It shows the Price positions for all products
```

1. Click on the red pen "Edit" button inside the third pie chart on "Price position" widget Action

2. Click on "Reset to default" button

Input data

Expected. "Widget was successfully updated" information message was displayed result. The widget editing popup is closed.

result

3. Third pie chart It shows the Price positions for all products

5 Step

Action Check result on database

Input data

```
In the fields params displayed JSON which displayed your filters:
           "origin": null,
           "brandslds": [],
           "labelsIds": [],
           "domainsIds": null,
           "dynamicTags": [],
           "mainFilters": null,
           "customLabels": [],
           "customFilters": null,
           "productTypesIds": [],
           "competitorsStockStatus": []
           },
           {
"origin": null,
           "brandslds": [],
           "labelsIds": [],
Expected "domainsIds": null, "dynamicTags": [],
result
           "mainFilters": null,
           "customLabels": [],
           "customFilters": null,
           "productTypesIds": [],
           "competitorsStockStatus": []
           },
           "origin": null,
           "brandslds": [],
           "labelsIds": [],
           "domainsIds": null,
           "dynamicTags": [],
           "mainFilters": null,
           "customLabels": [],
           "customFilters": null,
           "productTypesIds": [],
           "competitorsStockStatus": []
           }
           j
```

Dashboard-> "Selected Domains" widget-> Three radio-buttons

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from minimum 15 different domains.
- 2. Dashboard page is opened.
- 3. "Selected Domains" widget is added and filtered with selected 10 domains from different domain zones.

Step	1
Action	On the "Selected Domains" widget click on the second radio button.
Input data	
	Widget display was changed from the first page (top 1-5 domains with the biggest number of matches) to the sec ond page (top 6-10).
Step	2
Action	On the "Selected Domains" widget click on the third radio button.
Input data	
	Widget display was changed from the second page (top 6-10 domains with the biggest number of matches) to the third page (top 10- 15).
Step	3
Action	On the "Selected Domains" widget click on the first radio button.
Input data	
	Widget display was changed from the third page (top 10-15) to the first page (top 1- 5 domains with the biggest n umber of matches).
Step	4
Action	Check the values with DB.

Input data select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_i ndex, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join products USE INDEX (PRIMARY) on `matchings`.`product_id` = `products`.`id` where matchings.account_id = account_id and `competitors`.`deleted_at` is null and `products`.`deleted_at` is null group by `matchings`.`domain_id` order by `count` desc limit 15

Expected The values are the same. result

Dashboard -> "Selected Domains" widget-> Filters-> Mix "Tags", "Dynamic tags" + the missing stock status of a competitor

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Залогинится как юзер или админ, подключенный к своему аккаунту. В аккаунте есть товары с матчами.

От домена "test.dk" - максимальное количество матчей в аккаунте. У конкурента статус запасов только "в наличии". Все продукты, у которых есть конкурент "test.dk", имеют динамический тег "dynamic_tag_1" и тег "tag_1" Страница "Dashboard" открыта.

Добавлен виджет "Selected domains"

Step	1
Action	Нажать красную кнопку «Редактировать» виджета "Top domains".
Input data	
Expected result	Отображается всплывающее окно "Choose what you want to see"
Step	2
Action	Нажать на поле «Теги» и выбрать тег " tag_1" и "dynamic_tag_1".
Input data	
Expected result	Имена " tag_1" и "dynamic_tag" отображаются в поле поиска «Tags».
Step	3
Action	Нажать "Next" кнопку
Input data	
Expected result	Отображается всплывающее окно "CHOOSE WHICH DOMAINS YOU WANT TO SEE".
Step	4
Action	Выбрать домен "test.dk"
Input data	
Expected result	Чекбокс домена стал активен, имя домена отобразилось под полем поиска.
Step	5

Action	Нажать "Next" кнопку
Input data	
Expected result	Отображается всплывающее окно "Competitor stock filter".
Step	6
Action	Выбрать "Out of stock" параметр и нажать "Save" кнопку
Input data	
Expected result	У выбранного параметра активирован чекбокс. Появилось информационное сообщение «Widget updated successfully" is displayed». Отображается сообщение "Nothing to show at this moment. Please, try again later" Данные не отображаются.

Dashboard -> "Selected Domains" widget-> To increase/to decrease arrows sorting

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User. Account has products with matches from minimum 15 different domains. Dashboard page is opened. "Selected Domains" widget is added. 15 required domains added

Steps to reproduce

Step 1

Action

On the "Selected Domains" hover over "Matches", to the right of matches click on the "Up" arrow. Click on the 3 ratio buttons to scroll through the pagination (all 15 domains)

Input data

Expected Arrow left "Up". Domains are displayed in order from the one with the fewest matches to the most. Pay attention result to the sequence of all 15 domains



Step 2

Click on the "Up" arrow to the right of "Matches".

Click on the 3 ratio buttons to scroll through the pagination (all 15 domains)

Input data

Action

Expected Arrow became "Down". Domains are displayed in order from the one with the most matches to the fewest. Pay atl result ention to the sequence of all 15 domains



Step 3

Action Click on "Down" arrow.

Input data	
Expected result	Arrow disappeared. Domains are displayed in order from the one with the most matches to the fewest (defolt state).
Step	4
Action	Similarly reproduce steps 1-3 with Index, Average diff. parameters
Input data	
Expected result	Similarly, the arrows change the display of domains according to the selected parameter (Index, Average diff.) To increase "Up" arrow to decrease "Down" arrow

Dashboard -> "Selected Domains" widget-> Check data in widget

Кейс предназначен для проверки данных в "Selected Domains" виджете.

Severity	Priority	Behavior	Туре
O Normal	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами с разных доменов (>= 15). 2. На Dashboard странице добавить "Selected Domains" виджет.
- 3. Перейти на Dashboard страницу.

Steps t	Steps to reproduce		
Step	1		
Action	Проверить отображение доменов в "Selected Domains" виджете.		
Input data			
Expecte result	d. 1. В виджете отображается 15 доменов по 5 доменов на каждой пагинации.		
Step	2		
Action	Проверить порядок отображения доменов.		
Input data			
Expecte result	${ m d}_1$. Порядок отображения доменов отсортирован от наибольшего количества матчей к меньшему.		
Step	3		
Action	Сравнить данные в "Selected Domains" виджете и в базе данных.		
Input data	select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competit ors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitors'.id` inner join domains on `competitors`.`domain_id` = `domains`.id` inner join products USE INDEX (PRIMARY) on `matchings`.`product_id` = `products`.`id` where matchings.account_id =account_idand `competitors`.`d eleted_at` is null and `products`.`deleted_at` is null group by `matchings`.`domain_id` order by `count` desc limit 15		
Expecte result	${ m cd}_1$. Результаты запроса в базе данных соответствуют отображению в "Selected Domains" виджете.		
Step	4		

Action

Отфильтровать "Selected Domains" виджет по любому бренду, "In stock" статусу и сравнить отфильтрова нные данные в виджете и в базе данных.

Input data

select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competit ors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (P RIMARY) on `matchings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join pr oducts USE INDEX (PRIMARY) on `matchings`.`product_id` = `products`.`id` where matchings.account_id =account_idand `competitors`.`d $\textbf{eleted_at` is null and `products`.`} \textbf{deleted_at` is null and (`products`.`} \textbf{brand_id` in (} \textbf{brand_id})) \textbf{ and (`competitors`.`} \textbf{availability` != 'out of } \textbf{of } \textbf$ stock' or `competitors`.`availability` is null) group by `matchings`.`domain_id` order by `count` desc limit 15

result

Expected 1. Результаты запроса в базе данных соответствуют отображению в "Selected Domains" виджете.

Step

5

Action

Сбросить "Selected Domains" виджет до дефолтного состояния, выбрать 10 доменов и сравнить отфильт рованные данные в виджете и в базе данных.

Input data

select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competit ors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (P RIMARY) on `matchings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join pr oducts USE INDEX (PRIMARY) on `matchings`.`product id` = `products`.`id` where matchings.account id =account idand`competitors`.`d eleted at' is null and 'products'. 'deleted at' is null and 'matchings'. 'domain id' in (domain id 1,domain id 2,domain id 3,dom ain_id_4,domain_id_5,domain_id_6,domain_id_7,domain_id_8,domain_id_9,domain_id_10) group by `matchings`.`domai n_id` order by `count` desc limit 15

result

Expected 1. Результаты запроса в базе данных соответствуют отображению в "Selected Domains" виджете.

Dashboard -> Selected Domains widget -> Don`t display SD matches

Мягко-удаленные матчи не отображаются в Selected Domains виджете

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	

Pre-conditions

- 1. В аккаунте есть товары с матчами.
- 2. С доменом **stark.dk** есть только один матч, он мягко-удален.
- 3. С доменом **proshop.dk** есть 5 матчей, один матч мягко-удален.
- 4. Открыть Dashboard страницу, добавить Selected domains виджет.

Step	1
Action	Нажать на "Edit" Selected domains виджет, перейти на вкладку с выбором доменов.
Input data	
Expected result	dl. Открылось окно "EDIT SELECTED DOMAINS WIDGET". 2. На вкладке с выбором доменов отображается поиск с выбором доменов.
Step	2
Action	 Нажать на поле поиска и раскрыть дропдаун с доменными зонами. Раскрыть доменные зоны. Обратить внимание на счетчики.
Input data	SELECT * FROM `matchings` where competitor_id IN (SELECT id FROM `competitors` WHERE `account_id` = account_id AND `domain_id` = doma
Expected result	1. Отображается список доменов, сгруппированных по доменным зонам. 2. В списке доменов не отображается домен stark.dk. 3. Отображается для домена proshop.dk отображается счетчик (4).
Step	3
Action	Выбрать домен proshop.dk, сохранить виджет.
Input data	
Expected result	 Попап редактирования виджета закрыт. Отобразилось сообщение "Widget was successfully updated". На виджете отображается только proshop.dk домен со счетчиком матчей - 4.

E2E: Dashboard -> "Selected Domains" widget-> Filters-> "Brands"

Display "Selected Domains" widget filtered by brands

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from minimum 15 different domains.
- 2. Some products with matches from selected domains belongs to "Test1" and "Test2" brands.
- 3. There are no Domain Groups created in the account.
- 4. Dashboard page is opened.
- 5. "Selected Domains" widget is added.

Steps to reproduce

data

Step	1
Action	On the "Selected domains" widget click on the "Edit" button.
Input data	
Expectoresult	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fil l out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons.
Step	2
Action	Filter "Selected Domains" widget by brands "Test 1", "Test2". Use search field.
Input data	
Expectoresult	ed. Dropdown with Brands was displayed. 2. Search found "Test1" Brand. 3. Selected 2 Brand's, checkboxes are filled with checkmarks.
Step	3
Action	 Click on the "Next" button. Click on the different domains zones, choose 8 domains by clicking on checkboxes. Don't select competitor's availability.
Input	

1. EDIT SELECTED DOMAINS WIDGET" filter's popup displayed with text "Choose what you want to see and fill out at least one category."

Expected². Below is the "Name:" input field.
"In stock", "Out of stock" checkboxes.

result

- 4. Dropdown with domains (domains belonging to the selected domain zone) displayed.
- 5. Selected checkboxes are filled with a checkmarks.
- 6. 8 domains were selected and displayed below domains dropdown.

4 Step

Action Click on "Save" button.

Input data

Expected. "EDIT SELECTED DOMAINS WIDGE: INC. 3 F.F..."

"Widget was successfully updated" message displayed. "EDIT SELECTED DOMAINS WIDGET" filter's popup closed. 3. Under the widget name applied brands are displayed.

Step 5

1. Pay attention to the "A" domain data, number of matches. Action

2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

Expected. Redirect to Product list page filtered by "Test1" and "Test2" Brands and "A" domain.

2. Number of matches are the same and "A" domain.

result

Number of matches are the same, only products from "Test1" and "Test2" Brands and matches from "A" domai n are displayed.

Step 6

Action Compare values on the "Selected Domains" widget with the DB.

Input data

select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((co mpetitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_i ndex, ROUND(AVG((competitors.clear price / competitors.quantity) / (products.clear price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matc hings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join products USE INDEX (PRIMARY) on `matchings`.`product_id` = `products`.`id` where matchings.acco unt_id = account_id and `competitors`.`deleted_at` is null and `products`.` brand id' in (selected brand id)) and 'matchings'.'domain id' in (selected domain id) group by 'matchin gs'.'domain id' order by 'count' desc limit 15

Expected Values are the same. result

E2E: Dashboard -> "Selected Domains" widget-> Filters->"Product Types"

Display "Selected Domains" widget filtered by Product types

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
_	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from minimum 15 different domains.
- ${\bf 2. \ \ Some\ products\ with\ matches\ from\ selected\ domains\ belongs\ to\ "Test1"\ and\ "Test2"\ Product\ types.}$
- 3. Dashboard page is opened.4. "Selected Domains" widget is added.

Step	1
Action	On the "Selected domains" widget click on the "Edit" button.
Input data	
Expecte result	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fil lout at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons.
Step	2
Action	 Filter "Selected Domains" widget by Product types "Test 1", "Test2". Use search field for one check.
Input data	
Expecte result	1. Dropdown with Product types was displayed. 2. Search found "Test1" Product types. 3. Selected 2 Product types, checkboxes are filled with checkmarks.
Step	3
Action	 Click on the "Next" button. Click on the different domains zones, choose 6 domains by clicking on checkboxes. Don't select competitor's availability.
Input data	

1. EDIT SELECTED DOMAINS WIDGET" filter's popup is displayed with text "Choose what you want to see and fill o ut at least one category.'

Expected². Below is the "Name:" input field.
3. "In stock" "Out of stock" checkly

result

- "In stock", "Out of stock" checkboxes.Dropdown with domains (domains belonging to the selected domain zone)
- 4. Selected checkboxes are filled with a checkmarks.
- 5. 8 domains were selected and displayed below domains dropdown.

4 Step

Action Click on "Save" button.

Input data

result

Expected. "EDIT SELECTED DOMAINS WIDGET" filter's popup closed. 2. "Widget was successfully updated" message displayed.

Step 5

Action

- 1. Pay attention to the "A" domain data, number of matches.
- 2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

Expected. Redirect to Product list page filtered by "Test1" Product type "A" domain.

2. Number of matches are the same, only products from "Test1" Product type and "A" domain are displayed. result

Step 6

Action Compare values on the "Selected Domains" widget with the DB.

Input data

select `domain` as `name`, `domains`.`id` as `domain id`, count(competitor id) as count, ROUND(AVG((com petitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind ex, ROUND(AVG((competitors.clear price / competitors.quantity) / (products.clear price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE INDEX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`co mpetitor_id` = `competitors`.`id` inner join domains on `competitors`.`id` and in a `domains`.`id` where pro ducts.account_id = **account_id** and `competitors`.`deleted_at` is null and ((products.product_type_id IN **prod** uct type id))) and `domains`.`id` in (selected domains id) and `products`.`deleted at` is null group by `do mains'.'id' order by 'count' desc limit 15

Expected Values are the same. result

E2E: Dashboard -> "Selected Domains" widget-> Filters-> "Tags"

Display "Selected Domains" widget filtered by Tags

SeverityPriorityBehaviorTypeO NormalO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. Account has products with matches from minimum 5 different domains.
- 2. Some products with matches from selected domains belongs to "Test1" and "Test2" tags.
- 3. Dashboard page is opened.
- 4. "Selected Domains" widget is added.

Steps	Steps to reproduce		
Step	1		
Action	On the "Selected domains" widget click on the "Edit" button.		
Input data			
Expecte result	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fil l out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons. 		
Step	2		
Action	Filter "Selected Domains" widget by tags "Test 1", "Test2". Use search field.		
Input data			
Expecte result	1. Dropdown with Tags was displayed. 2. Search found "Test1" Tag. 3. Selected 2 Tags, checkboxes are filled with checkmarks.		
Step	3		
Action	 Click on the "Next" button. Click on the different domains zones, choose 3 domains by clicking on checkboxes. Don't select competitor's availability. 		
Input data			

1. EDIT SELECTED DOMAINS WIDGET" filter's popup is displayed with text "Choose what you want to see and fill o ut at least one category.'

Expected². Below is the "Name:" input field.
social and a state of the category?
In stock, "Out of stock," checkboxes.

result

- 4. Dropdown with domains (domains belonging to the selected domain zone) was displayed.
- 5. Selected checkboxes are filled with a checkmarks.
- 6. 3 domains were selected and displayed below domains dropdown.

4 Step

Action Click on "Save" button.

Input data

Expected. "EDIT SELECTED DOMAINS WIDGET" filter's popup closed.

2. "Widget was successfully updated" message displayed. result

5 Step

Action

- 1. Pay attention to the "A" domain data, number of matches.
- 2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

result

Expected. Redirect to Product list page filtered by "Test1" and "Test2" Tags and "A" domain.

2. Number of matches are the same, only products from "Test1" and "Test2" Tags and matches from "A" domain are displayed.

6 Step

Compare values on the "Selected Domains" widget with the DB. Action

Input data

select `domain` as `name`, `domains`.`id` as `domain id`, count(competitor id) as count, ROUND(AVG((com petitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind ex, ROUND(AVG((competitors.clear price / competitors.quantity) / (products.clear price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE INDEX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`co mpetitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where pro ducts.account_id = account_id and `competitors` `deleted_at` is null and ((exists (select * from `products_pro duct_labels` where `products`.`id` = `products_product_labels`.`product_id` and product_label_id IN (product_label_id,product_label_id)))) and `domains`.`id` in (selected_domains_id) and `products`.`deleted_at` is n ull group by `domains`.`id` order by `count` desc limit 15

Expected Values are the same. result

E2E: Dashboard -> "Selected Domains" widget-> Filters-> "Dynamic tags" parameters

SeverityPriorityBehaviorTypeO NormalO MediumPositiveFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. There are products with matches, different tags, dynamic tags in the account.
- 2. From the domain" test.dk" the maximum number of matches in the account with the status of a competitor "in stock" and "out of stock".
- 3. All products that have competitor "test.dk" have dynamic tag "test" applied to them.
- 4. Dashboard page is opened.
- 5. "Selected domains" widget is added.

Steps to reproduce

Step	1
Action	Click on the red pen "Edit" button on "Selected domains" widget
Input data	
Expected result	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see an d fill out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons.
Step	2
Action	 Click on the "Tags" field. Select the dynamic tags "test" (or write the dynamic tag name in the search field and select it by check).
Input data	
Expected result	 Dropdown with Tags was displayed. Search found "Test" Dynamic tag. Selected Dynamic tag checkbox is filled with checkmark.
Step	3
Action	Click on the "Next" button.
Input data	
Expected result	 EDIT SELECTED DOMAINS WIDGET" filter's popup displayed with text "Choose what you want to see and fill out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes.

4. Dropdown with domains (domains belonging to the selected domain zone) displayed.

Step

Action 2. "In stock" and "Out of stock" competitors stock statuses. Input data Expected 1. A checkmark appears next to the selected domain. 2. "In stock" and "Out of stock" filled with checkmarks. result 5 Step Click on the "Save" button Action Input data 1. The informational message "Widget updated successfully" is displayed. Expected 2. Under the widget name, the applied dynamic tag and the status of the competitor are displayed. result 6 Step Follow the arrow button domain near "test.dk" on the "Top domains" widget to page "Product list" Action Input data Expected Filtered products by "Top domains" widget are displayed result 7 Step Open in the next tab "Product list", select the same dynamic tag "test", domain " test.dk" and competitor Action statuses in the filters Input data 1. Filtering results from "Top domains" widget and "Product list" are the same. Expected result 2. The number of matches on the widget and on the "Product list" are the same.

1. Choose "test.dk" domain.

E2E: Dashboard -> "Selected Domains" widget-> Filters-> Mix "Tags" and "Dynamic tags"

SeverityPriorityBehaviorTypeO NormalO MediumPositiveRegressionMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

Залогинится как юзер или админ, подключенный к своему аккаунту.

В аккаунте есть товары с матчами.

От домена "test.dk" - максимальное количество матчей в аккаунте со статусом конкурента "в наличии" и "нет в наличии".

Все продукты, у которых есть конкурент "test.dk", имеют динамический тег "dynamic_tag_1" и тег "tag_1" Страница "Dashboard"открыта.

Добавлен виджет "Selected domains"

Step	1
Action	Нажать красную кнопку «Редактировать» виджета "Selected domains"
Input data	
Expected result	Отображается всплывающее окно "Choose what you want to see"
Step	2
Action	Нажать на поле «Теги» и выбрать тег " tag_1" и "dynamic_tag_1".
Input data	
Expected result	Имена " tag_1" и "dynamic_tag" отображаются в поле поиска «Tags».
Step	3
Action	Нажать "Next" кнопку
Input data	
Expected result	Отображается всплывающее окно "CHOOSE WHICH DOMAINS YOU WANT TO SEE".
Step	4
Action	Выбрать домен "test.dk"
Input data	

Expected result	Чекбокс домена стал активен, имя домена отобразилось под полем поиска.
Step	5
Action	Нажать "Next" кнопку
Input data	a
Expected result	Отображается всплывающее окно "Competitor stock filter".
Step	6
Action	Выбрать "In stock" и "Out of stock" параметр и нажать "Save" кнопку
Input data	a
Expected result	У выбранных параметров активирован чекбокс. Появилось информационное сообщение «Widget updated successfully" is displayed».
Step	7
Action	Перейти по кнопке "arrow" домена "test.dk" на страницу "Product list"
Input data	a
Expected result	Отображается отфильтрованные товары тегу " tag_1" и динамическому тегу "dynamic_tag_1", домену "tes t.dk", а также по выбранному сток статусу конкурента.
Step	8
Action	Открыть в соседней вкладке "Product list", выбрать в фильтрах тот же динамический тег "dynamic_tag_ 1", тег " tag_1", домен "test.dk" и сток статус конкурента.
Input data	a
Expected result	Результаты перехода по гиперссылке из виджета "Top domains" и на странице "Product list" совпадают. Количество матчей и данные на виджете и на "Product list" - одинаковое.

E2E: Dashboard -> Selected Domains widget -> Filters -> "Imported tags"

Display "Selected Domains" widget filtered by Imported Tags

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported_tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	

Pre-conditions

- 1. Account has products with matches from minimum 5 different domains.
- 2. Some products with matches from selected domains belongs to "Test1" and "Test2" imported tags.
- 3. Dashboard page is opened.
- 4. "Selected Domains" widget is added.

1. Click on the "Next" button.

3. Don't select competitor's availability.

Action

Steps to reproduce

Step	
Action	On the "Selected domains" widget click on the "Edit" button.
Input data	
Expecte result	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fill out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons.
Step	
Action	. Filter "Selected Domains" widget by imported tags "Test 1", "Test2". . Use search field.
Input data	
Expecte result	 Dropdown with imported tags was displayed. Search found "Test1" imported tag. Selected 2 imported tags, checkboxes are filled with checkmarks.
Step	

2. Click on the different domains zones, choose 3 domains by clicking on checkboxes.

Input data

- 1. EDIT SELECTED DOMAINS WIDGET" filter's popup is displayed with text "Choose what you want to see and fill o ut at least one category.'
- Expected². Below is the "Name:" input field.
 "In stock", "Out of stock" checkboxes.

result

- 4. Dropdown with domains (domains belonging to the selected domain zone) was displayed.
- 5. Selected checkboxes are filled with a checkmarks.
- 6. 3 domains were selected and displayed below domains dropdown.

Step 4

Action Click on "Save" button.

Input data

Expected. "EDIT SELECTED DOMAINS WIDGET" filter's popup closed.

2. "Widget was successfully updated" message displayed

Step 5

Action

- 1. Pay attention to the "A" domain data, number of matches.
- 2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

result

Expected. Redirect to Product list page filtered by "Test1" and "Test2" imported tags and "A" domain.

Number of matches are the same only products for "Test2" imported tags and "A" domain.

Number of matches are the same, only products from "Test1" and "Test2" imported tags and matches from "A" domain are displayed.

Step 6

Action Compare values on the "Selected Domains" widget with the DB.

Input data

select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((com petitors.clear price / competitors.quantity) / (products.clear price / products.quantity) * 100), 2) as domain ind ex, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average deviation from `products` inner join matchings USE INDEX (matchings 1 product id) on `matchings`.`product id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`co mpetitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where pro $\label{local_ducts} $$\operatorname{ducts.account_id} = <\operatorname{span} \ style="\operatorname{color:var(--md-font-color-failed)"} > \operatorname{account_id} </\operatorname{span} > \operatorname{and} \ \operatorname{competitors}. \ \ \operatorname{del} = \operatorname{ducts.account_id} </\operatorname{span} > \operatorname{and} \ \ \operatorname{competitors}. \ \ \operatorname{del} = \operatorname{duct_imported} = \operatorname$ _tags`.`product_id` and imported_tag_id IN (imported_tags_ids)))) and `domains`.`id` in (domain ids) and `p roducts`.`deleted_at` is null group by `domains`.`id` order by `count` desc limit 15;

Expected_{Values} are the same. result

E2E: Dashboard -> "Selected Domains" widget-> Filters-> "Labels"

Display "Selected Domains" widget filtered by Labels

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone			
Milestone	Automation	Status	Is Muted

Pre-conditions

- 1. Account has products with matches from minimum 5 different domains.
- ${\hbox{\bf 2. Some products with matches from selected domains belongs to "Test1" and "Test2" labels.}\\$
- 3. Dashboard page is opened.4. "Selected Domains" widget is added.

Steps to reproduce		
Step	1	
Action	On the "Selected domains" widget click on the "Edit" button.	
Input data		
Expecte result	 "EDIT SELECTED DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fil l out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Next" buttons. 	
Step	2	
Action	Filter "Selected Domains" widget by labels "Test 1", "Test2". Use search field.	
Input data		
Expecte result	1. Dropdown with Labels was displayed. 2. Search found "Test1" Label. 3. Selected 2 Label, checkboxes are filled with checkmarks.	
Step	3	
Action	 Click on the "Next" button. Click on the different domains zones, choose 3 domains by clicking on checkboxes. Don't select competitor's availability. 	
Input data		

1. EDIT SELECTED DOMAINS WIDGET" filter's popup is displayed with text "Choose what you want to see and fill o ut at least one category."

Expected². Below is the "Name:" input field.
3. "In stock", "Out of stock" checkboxes.

result

- 4. Dropdown with domains (domains belonging to the selected domain zone) was displayed.
- 5. Selected checkboxes are filled with a checkmarks.
- 6. 3 domains were selected and displayed below domains dropdown.

4 Step

Action Click on "Save" button.

Input data

Expected. "EDIT SELECTED DOMAINS WIDGET" filter's popup closed.

2. "Widget was successfully updated" message displayed.

5 Step

Action

- 1. Pay attention to the "A" domain data, number of matches.
- 2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

result

Expected. Redirect to Product list page filtered by "Test1" and "Test2" Labels and "A" domain.

2. Number of matches are the same, only products from "Test1" and "Test2" Labels and matches from "A" domain are displayed.

6 Step

Compare values on the "Selected Domains" widget with the DB. Action

ex, ROUND(AVG((competitors.clear price / competitors.quantity) / (products.clear price / products.quantity) * Input data

100) - 100, 2) as average_deviation from `products` inner join matchings USE INDEX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`co mpetitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where pro ducts.account_id = account_id and `competitors` `deleted_at` is null and ((custom_labels_data->"\$.custom_la bel_0" = "**Test1**" or custom_labels_data->"\$.custom_label_1" = "**Test1**" or custom_labels_data->"\$.custom_label_2" = "**Test1**" or custom_labels_data->"\$.custom_label_3" = "**Test1**" or custom_labels_data->"\$.custom_label I_4" = "Test1")) and `domains`.`id` in (selected_domains_id) and `products`.`deleted_at` is null group by `d omains`.`id` order by `count` desc limit 15

select `domain` as `name`, `domains`.`id` as `domain id`, count(competitor id) as count, ROUND(AVG((com petitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind

Expected Values are the same. result

E2E: Dashboard -> "Selected Domains" widget-> Filters-> "Competitor Stock Filter"

SeverityPriorityBehaviorType- Not setO MediumNot setFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. Account has products with matches from 6 different domains.
- 2. "A" domain is selected on widget, has "In stock" and " Out of stock" products.
- 3. Dashboard page is opened.
- 4. "Selected Domains" widget is added.

Steps to reproduce

Step 1

Action Compare values on the "Selected Domains" widget with the DB.

Input data

select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind ex, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE INDEX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where products.account_id = account_id and `competitors`.`deleted_at` is null and (`competitors`.`availability` != 'out of stock' or `competitors`.`availability` is null) and `domains`.`id` in (selected_domains_id) and `products`.` deleted_at` is null group by `domains`.`id` order by `count` desc limit 15

Expected Values are the same. result

Step 2

Action Compare values on the "Selected Domains" widget with the DB.

Input data

select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind ex, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE INDEX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where products.account_id = account_id and `competitors`.`deleted_at` is null and (`competitors`.`availability` = 'out of stock') and `domains`.`id` in (selected_domains_id) and `products`.`deleted_at` is null group by `domains`.`id` order by `count` desc limit 15

Expected Values are the same. result

Step 3

Filter "Selected Domains" widget by:

Action

- 5 domains from different domain zones;
- "In stock" competitors filter parameter;
- do not select brand, product type, tag, label.

Input data

result

- 1. "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
- 2. Choose nothing= choose all products.
- 3. "Choose which domains you want to see" popup is displayed.

Expected. Dropdown with domains (domains belonging to the selected domain zone) was displayed. Selected checkboxes are filled with a checkmark.

- 6. "Competitor stock filter" popup is displayed. 7. A checkmark appears next to the "In stock" option.
- 8. Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.
- 9. Pay atention to the 5 Selected domains.

4 Step

Action

Pay attention to the "A" domain data, number of matches. Opposite the "A" domain, click on the red arrow to th e "Right".

Input data

Expected. Redirect to Product list page filtered by "A" domain, all matches are in stock.

2. Pay attention to the number of matches, only matches In stock from "A" domain must be displayed. result

Step 5

On the new tab filter "Selected Domains" widget by:

Action

- same 5 domains from different domain zones;
- "Out of stock" competitors filter parameter;
- do not select brand, product type, tag, label.

Input data

1. Redirect to Product list page filtered by "A" domain, all matches are Out of stock.

Expected2. Pay attention to the number of matches, only matches Out of stock from "A" domain must be displayed.

result

3. Compare Product List pages with In stock from "A" domain and Out of stock from "A" domain- our products are different.

E2E: Dashboard ->"Selected Domains" widget->Changing the widget`s name

Setting a custom name of Selected Domains widget

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from minimum 5 different domains.
- 2. "Selected Domains" widget is added.
- 3. Open the Dashboard page.

Step	1
Action	On the Selected Domains widget click on the "Edit" red pen icon.
Input data	
Expected result	"EDIT SELECTED DOMAINS WIDGET" is displayed with: "Choose what you want to see and fill out at least one category." message. "Name:" input field. "In stock" and "Out of stock" check boxes. Brand, Product type, Tag, Label drop downs. "Reset to default" and "Next" buttons. red cross "Cancel".
Step	2
Action	Enter the new widgets name "Test" into input field.
Input data	Test
Expected result	The new widgets name "Test" is displayed in the input field
Step	3
Action	 Click on the "Next" button. Click on "Save".
Input data	
Expected result	 The tab with domains is displayed. Editing popup was closed. "Widget was successfully updated" informational message was displayed. Pay attention to the new widgets name "SELECTED DOMAINS - Test"

Dashboard -> Gross margin widget -> Mix "Tags", "Dynamic tags" parameters + the missing own stock status

Severity **Priority Behavior Type** O Normal O Medium Positive **Functional** Milestone **Automation** Status Is Muted December 1.0 (2022) Manual Actual

Pre-conditions

- 1. К группе продуктов применен tag_1 и dynamic_tag.
- 2. У товаров статус запасов: In stock.
- 3. У товаров есть атрибут cost price.
- 4. Страница "Dashboard" открыта.
- 5. Добавлен виджет "Gross margin".

Steps to reproduce

1
Нажать красную кнопку «Редактировать» внутри одной из круговых диаграмм виджета "Gross margin".
1. Отображается окно "Choose what you want to see and fill out at least one category". ② Вверху окна отображается фильтр по собственному состоянию запасов. 3. Под фильтром состояния запасов отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
2
Нажать на поле «Теги» и выбрать тег " tag_1" и "dynamic_tag_1".
1. Имена " tag_1" и "dynamic_tag" отображаются в поле поиска «Tags», а также под ним с красным крестом слева.
3
Нажать "Next" кнопку

result

- Expected 3. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do mains, all domains are selected and new domains will be auto selected" message".
 - 4. Кнопки "Resert to default", "Back" и "Save"активны.

Action	Выбрать "Out of stock","Preorder	" параметры и нажать "Save" кнопку
--------	----------------------------------	------------------------------------

Input data

1. У выбранного параметра активирован чекбокс.

Expected: Появилось информационное сообщение «Widget updated successfully" is displayed». result
3. Отображается сообщение "No data", диаграмма - кольцо не отображается.
4. Среднее значение марджина отображается 0%.

Dashboard -> Gross margin widget -> Several parameters

Setting "Gross margin" widget by several parameters

Behavior Severity **Priority Type** O Normal O Medium Not set Regression Milestone **Automation Status** Is Muted December 1.0 (2022) Manual Actual No

Pre-conditions

- 1. There are products with cost price atribute, different brands, tags, dt, product types, labels (custom labels in the account.
- 2. Product "Test" has High margin, "out of stock" own status and belongs to:- "A" brand", "B" product type, "C" tag,"D"custom label, "F" Dynamic tag.
- 3. The product has "Test" a match from the domain "test.dk"
- 4. Dashboard page is opened.

Steps to reproduce

Step

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Gross margin" widget
Input data	
Expecte result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your own stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Brands" field and select "A" brand (or write the brand name "A" in the search field and select it b y check)
Input data	
Expecte result	d 1. The brand's name "A" is displayed in the "Brands" search field, as well as under the search field with a cross o n the left.
Step	3
Action	Click on the "Product types" field and select "B" product type (or write the product type name in the search fie Id and select it by check)
Input data	
Expecte result	d 1. The product type "B"name is displayed in the "Product types" search field, as well as under the search field with hacross on the left.

Action	Click on the "Tags" field and select "C" tag (or write the tag name in the search field and select it by check)
Input data	
Expected result	11. The tag's name "C" is displayed in the "Tags" search field, as well as under the search field with a cross on the left.
Step	5
Action	Click on the "Tags" field and select "F" dynamic tag (or write the dt name in the search field and select it by ch eck)
Input data	
Expected result	11. The dt`s name "F" is displayed in the "Tags" search field, as well as under the search field with a cross on the left.
Step	6
Action	Click on the "Labels" field and select "D" label (or write the label name in the search field and select it by check)
Input data	
Expected result	1. The label's name "D" is displayed in the "Labels" search field
Step	7
Action	Click on the "Next" button
Input data	
Expected result	 Below the name of the popup, you can see a filter by your own stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message. The buttons "Resert to default", "Back" and "Save" are active.
Step	8
Action	Choose "Out of stock" parameter
Input data	
Expected result	1. A checkmark appears next to the selected option
Step	9
Action	Choose a competitor domain "test.dk".

Input data	
Expected result	"Please be aware that if you don't select any domains, all domains are selected and new domains will be auto sel ected" message changed to counter "Selected 1 domain(s)"
Step	10
Action	Click on the "Save" button
Input data	
Expected	 The informational message "Widget updated successfully" is displayed. Notice the edited pie chart. It displays the Brands, Tag's, Product types, Labels, Dynamic tags name, quantity of goods in accordance with their gross margin, avg value gross margin and own stock status
Step	11
Action	Click on the high gross margin segment parameter

Input data

Expected 1. Product list with selected parameters is open. result 2. The "Test" product is displayed.

Dashboard -> Gross margin widget -> Display applied filters in hover

SeverityPriorityBehaviorTypeO NormalO MediumPositiveRegressionMilestoneAutomationStatusIs MutedDecember 1.0 (2022)ManualActualNo

Pre-conditions

- 1. У продукта №1 есть бренд "А", продукт тип "В", Тег "one", DT "DT_1", Label "Custom", статус "в наличии". Также у продукта есть матч с домена "test.dk"
- 2. Страница "Dashboard" открыта.
- 3. Добавлен виджет "Gross margin".

Step	1
Action	Нажать красную кнопку «Редактировать» внутри одной из круговых диаграмм виджета "Gross margin".
Input data	
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". cd. Вверху окна отображается фильтр по собственному состоянию запасов. 3. Под фильтром состояния запасов отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
Step	2
Action	Нажать на поле Теги» и выбрать тег " one" и динамический тег "DT_1"
nput data	
Expecte esult	dl. Имена " tag_1" и "dynamic_tag" отображается в поле поиска «Tags», а также под поиском с красным кр стом слева.
Step	3
Action	Нажать на поле «Brands» и выбрать бренд "А"
nput data	
Expecte result	d. Имя бренда "А" отображается в поле поиска «Brands», а также под поиском с красным крестом слева.
Step	4
Action	Нажать на поле «Product Types» и выбрать продукт тип "В"

Input data	
Expecte result	dl. Имя типа "В" отображаются в поле поиска «Product Types», а также под поиском с красным крестом сле ва.
Step	5
Action	Нажать на поле «Labels» и выбрать лейбл "Custom"
Input data	
Expecte result	^{cd} l. Имя лейбла "Custom" отображается в поле поиска «Labels», а также под поиском с красным крестом сле ва.
Step	6
Action	Нажать "Next" кнопку
Input data	
Expecte result	 Вверху окна отображается фильтр по собственному состоянию запасов. Отображается фильтр доменов. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do mains, all domains are selected and new domains will be auto selected" message". Кнопки "Resert to default", "Back" и "Save"активны.
Step	7
Action	Выбрать "In stock" статус Выбрать домен "test.dk".
Input data	
Expecte result	 У параметра сток статуса активирован чекбокс. Чекбокс домена также активирован. Домен "test.dk" отображается внизу поля поиска доменов с красным крестом. "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto s elected" сообщение сменилось на счетчик "Selected 1 domain(s)"
Step	8
Action	Сохранить изменения.
Input data	
Expecte result	rdl. Появилось информационное сообщение «Widget updated successfully" is displayed». 2. Отображается внутри диаграммы "1 products" и среднее (avg%) значение марджина.
Step	9
Action	Навести курсором на названия примененных фильтров внизу диаграммы

Input data

Отображается тултип, в котором указано, что выбрано:

- Сток статус: "In stock" Бренд "А" Expected Продукт тип "В" result Ter " one"

- Динамический тег "DT_1"Лейбл "Custom"

Выбранный домен "test.dk" НЕ отображается в тултипе.

Dashboard -> Gross margin widget -> General agenda

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
December 1.0 (2022)	Manual	Actual	No

Pre-conditions

У товаров есть атрибут cost price.

В настройках установлены такие параметры (Account setup)

- Low <20%
- Semi low 40%-20%
- Regular 60%-40%
- Semi high 80%-60%
- High >80%

Страница "Dashboard"открыта. Виджет "Gross margin" добавлен.

Step	1
Action	Навести курсором на "Low" позицию вверху диаграммы
Input data	
Expected result	 На всех трех диаграммах одновременно подсветились "Low" секции с процентным соотношением про дуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	2
Action	Навести курсором на "Semi low" позицию вверху диаграммы
Input data	
Expected result	 На всех трех диаграммах одновременно подсветились "Semi low" секции с процентным соотношение м продуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	3
Action	Навести курсором на "Regular " позицию вверху диаграммы
Input data	
Expected result	 На всех трех диаграммах одновременно подсветились "Regular" секции с процентным соотношением продуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	4

Action	Навести курсором на "Semi high" позицию вверху диаграммы
Input data	
Expected result	 На всех трех диаграммах одновременно подсветились "Semi hight" секции с процентным соотношени ем продуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"
Step	5
Action	Навести курсором на "High" позицию вверху диаграммы
Input data	
Expected result	 На всех трех диаграммах одновременно подсветились "High" секции с процентным соотношением пр одуктов от общей суммы. Количество продуктов на диаграмме "All products" совпадает со значениями в "Account setup"

Dashboard -> Gross margin widget -> Average gross margin value

Severity **Priority Behavior Type** O Medium O Normal Positive Regression Milestone **Automation Status** Is Muted December 1.0 (2022) Manual Actual No

2 2

Pre-conditions

- 1. У товаров есть атрибут cost price.
- 2. У продукта №1 gross margin = 56.78, продукта №2 78.22, продукта №3 = 44.32
- 3. К товарам №1- №3 применен тег "avg tag".
- 4. Страница "Dashboard"открыта.
- 5. Виджет "Gross margin" добавлен.

Steps to reproduce

Step	1
Action	Открыть режим редактирования виджета.
Input data	
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". cd. Вверху окна отображается фильтр по собственному состоянию запасов. 3. Под фильтром состояния запасов отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
Step	2
Action	Нажать на поле «Tags» и выбрать тег "avg tag".
Input data	
Expecte result	ed. 1. Имя тега отображается в поле поиска «Tags», а также под поиском с красным крестом слева.
Step	3
Action	Нажать "Next" кнопку
Input data	
Expecte result	1. Вверху окна отображается фильтр по собственному состоянию запасов. 2. Отображается фильтр доменов. 3. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any do

mains, all domains are selected and new domains will be auto selected" message".

4. Кнопки "Resert to default", "Back" и "Save" активны.

Action Не выбирая статусы запасов и домены, нажать на "Save" кнопку.

Input data

Expected. Появилось информационное сообщение "Widget updated successfully".

2. AVG значение внутри круговой диаграммы = 59,77%(gross_margin_product_№1 + gross_margin_product_№ 2 ..+ N) / count_products.

Dashboard -> Gross margin widget -> Redirecting to the Product list page

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
December 1.0 (2022)	Manual	Actual	No

Pre-conditions

У товаров есть атрибут cost price.

В настройках установлены такие параметры (Account setup)

- Low <20%
- Semi low 40%-20%
- Regular 60%-40%
- Semi high 80%-60%
- High >80%

B User setup включено отображение атрибута "Gross margin".

Страница "Dashboard"открыта. Виджет "Gross margin" добавлен, в нем отображается 5 секций.

Step	1
Action	Нажать на "Low" секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Low " диаграммы "All products" совпадает со значениями в "Product list". Значения "Gross margin", которые отображаются на карточках продуктов, соответствуют диапазон у "Low"
Step	2
Action	1. Открыть Dashboard страницу. 2. Нажать на "Semi low" секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Semi Low" диаграммы "All products" совпадает со значениями в "P roduct list". Значения "Gross margin", которые отображаются на карточках продуктов, соответствуют диапазон у "Semi low"
Step	3
Action	1. Открыть Dashboard страницу. 2. Нажать на "Regular " секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Regular диаграммы "All products" совпадает со значениями в "Product list". Значения "Gross margin", которые отображаются на карточках продуктов, соответствуют диапазон у "Regular"

Step	4
Action	1. Открыть Dashboard страницу. 2. Нажать на "Semi high" секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "Semi high" диаграммы "All products" совпадает со значениями в "Product list". Значения "Gross margin", которые отображаются на карточках продуктов, соответствуют диапазон у "Semi high"
Step	5
Action	1. Открыть Dashboard страницу. 2. Нажать на "High" секцию диаграммы "All products"
Input data	
Expected result	 Количество продуктов на секции "High" диаграммы "All products" совпадает со значениями в "Product list". Значения "Gross margin", которые отображаются на карточках продуктов, соответствуют диапазон у "High"

Dashboard->"Gross margin" widget->Check data in widget

Тест кейс проверяет отображение данных и выборке в виджете Gross margin сравнивая с базой данных

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открыта страница Dashboard {Host}/
- 2. Виджет Gross margin добавленный в аккаунте и находится в дефолтном состоянии.
- 3. В настройках градусника Gross margin установленные такие периоды High = your_high
 - Regular Between =your_regular_1, and =your_regular_2 Low = your_low
- 4. В аккаунте есть продукты с cost_price и price

Steps to reproduce

Step	1
Action	Обратить внимание на значения в первой диагреме виджета
Input data	
Expecte result	 Отображается диагрмама с тремя периодами High, Regular, Low В периоде High отображается Н-продуктов В периоде Regular отображается R-продуктов В периоде Low отображается L-продуктов Где H,R,L количество Общее количество продуктов = количеству продуктов, у которых есть все Gross margin, кроме No gross margin.
Step	2
Action	Проверить результаты в базе данных с значениями в диаграме
Input data	SELECT COUNT(stored_properties ->>'\$."gross_margin_color"') as count_products, stored_properties ->>'\$."gross_margin_color"' as period FROM `sphinx_products` WHERE `account_id` = account_id and stored_properties ->>'\$."gross_margin_color"' !='null' GROUP BY stored_properties ->>'\$."gross_margin_color"';
Expecte	1. количество продуктов в базе соотвецтвует диаграме в виджете: ed 1. Н-продуктов - High = count - green-margin

result

- 1. H-продуктов High = count green-margin
- 2. R-продуктов Regular = count gray-margin
- 3. L-продуктов Low = count red-margin

Step 3

Action Обратить внимание на значения в второй диагреме виджета

Input data

- 1. Отображается диагрмама с тремя периодами High, Regular, Low
- 2. В периоде High отображается H-продуктов
- Expected 4. В периоде Regular отображается R-продуктов 4. В периоде Low отображается L-продуктов

result

- 5. Где H,R,L количество
- 6. Общее количество продуктов = количеству продуктов бренда с самым большим количеством продуктов в аккаунте, у которых есть все Gross margin, кроме No gross margin.

4 Step

Action Проверить результаты в базе данных с значениями в диаграме

Input data

SELECT COUNT(stored properties ->>'\$."gross margin color"') as count products, stored properties ->>'\$."gro ss margin color" as period FROM `sphinx products` WHERE brand id =brand_id and `account_id` = account_id </sp an>and stored_properties ->>'\$."gross_margin_color"' !='null' GROUP BY stored_properties ->>'\$."gross_margi n color";

1. количество продуктов в базе соотвецтвует диаграме в виджете:

result

- Expected 1. H-продуктов High = count green-margin
 - 2. R-продуктов Regular = count gray-margin
 - 3. L-продуктов Low = count red-margin

Step 5

Action Обратить внимание на значения в третей диагреме виджета

Input data

- 1. Отображается диагрмама с тремя периодами High, Regular, Low
- 2. В периоде High отображается Н-продуктов

Expected. В периоде Regular отображается R-продуктов

result

- 4. В периоде Low отображается L-продуктов
- 5. Где H,R,L количество
- 6. Общее количество продуктов = количеству продуктов продукт типа с самым большим количеством про дуктов в аккаунте, у которых есть все Gross margin, кроме No gross margin.

Step 6

Проверить результаты в базе данных с значениями в диаграме Action

Input data

SELECT COUNT(stored properties ->>'\$."gross margin color"') as count products, stored properties ->>'\$."gro ss_margin_color" as period FROM `sphinx_products` WHERE product_type_id= product type id and`account id` = ac count_id and stored_properties ->>'\$."gross_margin_color"' !='null' GROUP BY stored_properties ->>'\$."gross_margin_color"'

1. количество продуктов в базе соотвецтвует диаграме в виджете:

result

- Expected 1. H-продуктов High = count green-margin
 - 2. R-продуктов Regular = count gray-margin
 - 3. L-продуктов Low = count red-margin

E2E: Dashboard -> Gross margin widget -> Brands parameters

Setting "Gross margin" widget by "Brands" parameters

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted

Pre-conditions

- Account has products with and without cost price, as well as products from different brands.
 Dashboard page is open with the "Gross margin" widget added.

Steps t	o reproduce
Step	1
Action	Click the red pen "Edit" button inside a pie chart on the "Gross margin" widget.
Input data	
Expecte result	1. "Choose what you want to see and fill out at least one category" popup is displayed. 2. The popup contains filters by stock status and product attributes. 3. "Reset to default" and "Next" buttons are active.
Step	2
Action	Click on the brands field, select a brand, or enter the brand name in the search field.
Input data	
Expecte result	${ m d}_{ m 1.}$ The selected brand is displayed in the Brands search field.
Step	3
Action	Click on the "Next" button.
Input data	
Expecte result	 Filters by stock status and domains are displayed. セ. A message "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" is shown at the bottom regarding domain selection. 3. "Reset to default", "Back", and "Save" buttons are active.
Step	4
Action	Choose "In stock" parameter.

Input data $\underset{\cdot\cdot}{\text{Expected}}_{1}. \ \ \text{A checkmark appears next to the selected option.}$ result 5 Step Action Choose a competitor domain that is a match for the product with the selected brand and own stock status. Input data Expected. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message changed to counter "Selected 1 domain(s)" message. result Step 6 Action Save changes. Input data 1. The informational message "Widget updated successfully" is displayed. Expected2. The pie chart displays brand name, quantity of goods with gross margin, avg gross margin, and own stock statu result s. 3. Products without the cost price attribute are excluded. 7 Step Action Follow the hyperlink on the "Gross margin" widget to page "Product list". Input data Expected 1. Filtered products by gross margin are displayed. result Step 8 Check results in DB. Action **SELECT** AVG(CAST(JSON UNQUOTE(JSON EXTRACT(stored properties, '\$.gross margin')) AS DECIMAL(10, 2))) as avg m GROUP_CONCAT(id) as product_id, SELECT COUNT(*) FROM sphinx_products AS sub WHERE account_id = <your_account_id> Input AND brand_id IN (<your_brand_id>) data AND availability NOT IN ('out of stock', 'preorder') AND JSON_UNQUOTE(JSON_EXTRACT(sub.stored_properties, '\$.gross_margin')) IS NOT NULL) as product_count FROM sphinx products WHERE account_id = <your_account_id> AND brand_id IN (<your_brand_id>) AND availability NOT IN ('out of stock', 'preorder') AND JSON_UNQUOTE(JSON_EXTRACT(stored_properties, '\$.gross_margin')) IS NOT NULL;

 $\begin{array}{l} \text{Expected}_{1}. \ \ \text{Results are the same}. \\ \text{result} \end{array}$

E2E: Dashboard -> Gross margin widget -> Product Types parameters

Setting "Gross margin" widget by "Product Types" parameters

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
December 1.0 (2022)	Manual	Actual	No

Pre-conditions

- 1. Account has products with attribute cost price and products not with this attribute.
- 2. There are products with different product type in the account.
- 3. Dashboard page is opened.4. "Gross margin" widget is added.

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Gross margin" widget
Input data	
Expecte result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your own stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Product types" field and select one of the product types (or write the product type name in the sea rch field and select it by check)
Input data	
Expecte result	d. The name of the product type is displayed in the Product types search field, as well as under the Product types search field with a cross on the left.
Step	3
Action	Click on the "Next" button
Input data	
Expecte result	 Below the name of the popup, you can see a filter by your own stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message. The buttons "Resert to default", "Back" and "Save" are active.

Action	Choose "In stock" (or "Out of stock" or "Preorder") parameter
Input data	
Expecte result	d 1. A checkmark appears next to the selected option
Step	5
Action	Choose a competitor domain that is a match for the product with the selected product types and own stock stat us.
Input data	
Expecte result	dl. "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto s elected" message changed to counter "Selected 1 domain(s)"
Step	6
Action	Save changes.
Input data	
Expecte result	 The informational message "Widget updated successfully" is displayed. In pie chart It displays the Product types name, quantity of goods in accordance with their gross margin, avg va lue gross margin and own stock status If some product of this product type does not have the cost price attribute, then such a product will not be take n into account in the widget, because it is impossible to calculate the gross margin.
Step	7
Action	Follow the hyperlink on the "Gross margin" widget to page "Product list"
Input data	
Expecte result	d 1. Filtered products by gross margin are displayed
Step	8
Action	Open in the next tab "Product list", select the same product type, gross margin and competitor domain, own sto ck status in the filters
Input data	
Expecte result	d 1. Filtering results from "Gross margin" widget and "Product list" are the same

E2E: Dashboard -> Gross margin widget -> Imported tags parameters

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	

Pre-conditions

- 1. В аккаунте есть продукт №1 и №2 с cost price атрибутом.
- У продукта №1 импортные теги test1, test2. У продукта №2 импортные теги test3, test4.
 На странице "Dashboard" добавлен "Price position" виджет.
 Открыть страницу "Dashboard".

Step	1
Action	Открыть режим редактирования одной из диаграм "Gross margin" виджета.
Input data	
Expecte result	d _{Открылся} "EDIT GROSS MARGIN WIDGET" попап.
Step	2
Action	Раскрыть фильтр "Tag" -> "Imported tags"
Input data	
Expecte result	d Отображаются импортные теги всех продуктов с cost price в аккаунте.
Step	3
Action	Выбрать импортный тег "test1"
Input data	
Expecte result	d Импортный тег "test1" отображается под фильтром "Tag"
Step	4
Action	Нажать "Next" кнопку.

Input data	
Expecte result	еd Отображается фильтр по доменам.
Step	5
Action	Нажать "Save" кнопку.
Input data	
Expecte result	edl. Отображается "Widget was successfully updated" сообщение. 2. В диаграмме отображается "1 products"
Step	6
Action	Выполнить запрос в БД.
Input data	select count(*) as Products from sphinx_products inner join product_imported_tags on sphinx_products.id = pro duct_imported_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id where imported_tags.name = ' imported_tags_name ' and sphinx_products.cost_price is not null and sphinx_products.account_id = account_id ;
Expecte result	еd Результат совпадает.

E2E: Dashboard -> Gross margin widget -> Labels parameters

Setting "Gross margin" widget by "Labels" parameters

Priority Behavior Severity **Type** O Normal O Medium Positive **Functional** Milestone Is Muted **Automation Status** December 1.0 (2022) Manual Actual No

3

Pre-conditions

- 1. Account has products with attribute cost price and products not with this attribute.
- 2. There are products with different labels in the account.
- 3. Dashboard page is opened.
- 4. "Gross margin" widget is added.

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Price position" widget
Input data	
Expecte result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your own stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Labels" field and select one of the labels (or write the label name in the search field and select it by check)
Input data	
Expecte result	d. The label's name is displayed in the "Labels" search field, as well as under the "Labels" search field with a cross on the left.
Step	3
Action	Click on the "Next" button
Input data	
Expecte result	 Below the name of the popup, you can see a filter by your own stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message. The buttons "Resert to default", "Back" and "Save" are active.

Action	Choose "In stock" (or "Out of stock" or "Preorder") parameter
Input data	
Expecte result	1. A checkmark appears next to the selected option
Step	5
Action	Choose a competitor domain that is a match for the product with the selected brand and own stock status.
Input data	
Expecte result	d"Please be aware that if you don't select any domains, all domains are selected and new domains will be auto sele cted" message changed to counter "Selected 1 domain(s)"
Step	6
Action	Save changes
Input data	
Expecte result	 The informational message "Widget updated successfully" is displayed. In pie chart It displays the Label's name, quantity of goods in accordance with their gross margin, avg value gross margin and own stock status If some product of this label does not have the cost price attribute, then such a product will not be taken into a ccount in the widget, because it is impossible to calculate the gross margin.
Step	7
Action	Follow the hyperlink on the "Gross margin" widget to page "Product list"
Input data	
Expecte result	1. Filtered products by gross margin are displayed
Step	8
Action	Open in the next tab "Product list", select the same label, gross margin and competitor domain, own stock status in the filters
Input data	
Expected result	d 1. Filtering results from "Gross margin" widget and "Product list" are the same

E2E: Dashboard -> Gross margin widget -> Tags parameters

Setting "Gross margin" widget by "Tags" parameters

Behavior Severity **Priority Type** O Normal O Medium Positive **Functional** Is Muted Milestone **Automation Status** December 1.0 (2022) Manual Actual No

Pre-conditions

1. Account has products with attribute cost price and products not with this attribute.

4. The buttons "Resert to default", "Back" and "Save" are active.

- 2. There are products with different tags in the account.
- 3. Dashboard page is opened.
- 4. "Gross margin" widget is added.

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Gross margin" widget
Input data	
Expecte result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your own stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Tags" field and select one of the tags (or write the tag name in the search field and select it by ch eck)
Input data	
Expecte result	ed1. The name of the tag is displayed in the Tags search field, as well as under the Tags search field with a cross on the left.
Step	3
Action	Click on the "Next" button
Input data	
Expecte result	 Below the name of the popup, you can see a filter by your own stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message.

Action	Choose "In stock" (or "Out of stock" or "Preorder") parameter
Input data	
Expected result	1. A checkmark appears next to the selected parameter
Step	5
Action	Choose a competitor domain that is a match for the product with the selected tags and own stock status.
Input data	
Expected result	"Please be aware that if you don't select any domains, all domains are selected and new domains will be auto sel ected" message changed to counter "Selected 1 domain(s)"
Step	6
Action	Save changes.
Input data	
Expected result	 The informational message "Widget updated successfully" is displayed. In pie chart It displays the Tag's name, quantity of goods in accordance with their gross margin, avg value gros s margin and own stock status If some product of this tags does not have the cost price attribute, then such a product will not be taken into ac count in the widget, because it is impossible to calculate the gross margin.
Step	7
Action	Follow the hyperlink on the "Gross margin" widget to page "Product list"
Input data	
Expected result	1. Filtered products by gross margin are displayed
Step	8
Action	Open in the next tab "Product list", select the same tag, gross margin and competitor domain, own stock status in the filters
Input data	
Expected result	1. Filtering results from "Gross margin" widget and "Product list" are the same

E2E: Dashboard -> Gross margin widget -> Dynamic tags parameters

SeverityPriorityBehaviorTypeO NormalO MediumPositiveFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. Account has products with attribute cost price and products not with this attribute.
- 2. There are products with different DT`s in the account.
- 3. Dashboard page is opened.
- 4. "Gross margin" widget is added.

Step	1
Action	Click on the red pen "Edit" button inside one of the pie charts on "Gross margin" widget
Input data	
Expecte result	 "Choose what you want to see and fill out at least one category" popup is displayed. Below the name of the popup, you can see a filter by your own stock status. Under the status stock filter, filters by product attributes are displayed. The buttons "Resert to default" and "Next" are active.
Step	2
Action	Click on the "Tags" field and select one of the dynamic tags (or write the dynamic tag name in the search field and select it by check)
Input data	
Expecte result	d1. The name of the dynamic tag is displayed in the Tags search field, as well as under the Tags search field with a cross on the left.
Step	3
Action	Click on the "Next" button
Input data	
Expecte result	 Below the name of the popup, you can see a filter by your own stock status. Domains filter is displayed. At the bottom of the popup is a "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message. The buttons "Resert to default", "Back" and "Save" are active.

Input data	
Expected result 1. A checkmark appears next to the selected parameter	
Step 5	
Action Choose a competitor domain that is a match for the product with the selected dynamic tags and own stock status.	t
Input data	
Expected "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto result ected" message changed to counter "Selected 1 domain(s)"	sel
Step 6	
Action Save changes.	
Input data	
 The informational message "Widget updated successfully" is displayed. In pie chart It displays the DT's name, quantity of goods in accordance with their gross margin, avg value g margin and own stock status If some product of this DT does not have the cost price attribute, then such a product will not be taken into ount in the widget, because it is impossible to calculate the gross margin. 	
Step 7	
Action Follow the hyperlink on the "Gross margin" widget to page "Product list"	
Input data	
Expected ₁ . Filtered products by gross margin are displayed result	
Step 8	
Action Open in the next tab "Product list", select the same dynamic tag, gross margin and competitor domain, own sock status in the filters	0
Input data	
Expected ₁ . Filtering results from "Gross margin" widget and "Product list" are the same result	

E2E: Dashboard -> Gross margin widget -> Mix "Tags" and "Dynamic tags" parameters

Severity **Priority Behavior Type** O Normal O Medium Positive **Functional** Milestone Automation Is Muted Status December 1.0 (2022) Manual Actual No

Pre-conditions

- 1. У товаров различный статус запасов: In stock, Out of stock, Preorder.
- 2. У товаров есть атрибут cost price.
- 3. К товарам применен tag_1 и dynamic_tag.
- 4. Страница "Dashboard"открыта.
- 5. Добавлен виджет "Gross margin".

Step	1
Action	1. Нажать красную кнопку «Редактировать» внутри одной из круговых диаграмм виджета "Gross margin ".
Input data	
Expecte result	1. Отображается окно "Choose what you want to see and fill out at least one category". etc. Вверху окна отображается фильтр по собственному состоянию запасов. 3. Под фильтром состояния запасов отображаются фильтры по атрибутам товара. 4. Кнопки "Resert to default" и "Next"активны.
Step	2
Action	1. Нажать на поле «Теги» и выбрать тег " tag_1" и "dynamic_tag_1".
Input data	
Expecte result	edl. Имена " tag_1" и "dynamic_tag" отображаются в поле поиска «Tags», а также под ним с красным крестог слева.
Step	3
Action	Нажать "Next" кнопку
Input data	
Expecte result	 Вверху окна отображается фильтр по собственному состоянию запасов. Отображается фильтр доменов. В нижней части всплывающего окна находится сообщение "Please be aware that if you don't select any domains, all domains are selected and new domains will be auto selected" message". Кнопки "Resert to default", "Back" и "Save"активны.

Action	1.	Выбрать "In stock" (или "Out of stock" или "Preorder") параметр и нажать "Save" кнопку
Input data		
Expecte result		У выбранного параметра активирован чекбокс. Появилось информационное сообщение «Widget updated successfully" is displayed».
Step	5	
Action	1.	Выбрать домен конкурента, который соответствует продукту с выбранными динамическими тегами и собственным статусом запаса.
Input data		
Expecte result	edī.	"Please be aware that if you don't select any domains, all domains are selected and new domains will be auto s elected" сообщение сменилось на счетчик "Selected 1 domain(s)"
Step	6	
Action	1.	Сохранить изменения.
Input data		
Expecte result	eď ^{2.}	Отображается информационное сообщение "Widget updated successfully". На круговой диаграмме отображается название Тега/ДТ, количество товаров в соответствии с их валово й прибылью, среднее значение валовой прибыли и состояние собственного запаса. Если какой-то товар этого Тега/ДТ не имеет атрибута cost price, то такой товар не будет учитываться в виджете, потому что невозможно рассчитать валовую прибыль.
Step	7	
Action	1.	Перейти по гиперссылке виджета "Gross margin" на страницу "Product list"
Input data		
Expecte result	edī.	Отображается отфильтрованные товары по гросс марджину, тегу " tag_1" и динамическому тегу "dyna mic_tag_1", а также по выбранному сток статусу собственного продукта и домену конкурента.
Step	8	
Action	1.	Открыть в соседней вкладке "Product list", выбрать в фильтрах тот же динамический тег "dynamic_tag _1", тег " tag_1", гросс марджин, сток статус своего продукта и домен конкурента.
Input data		
Expecte result	ed _{1.}	Результаты перехода по гиперссылке из виджета "Gross margin" и на странице "Product list" совпадают.

E2E: Dashboard -> Gross margin widget -> Removed filters

Severity **Priority Behavior Type** O Normal O Medium Positive **Functional** Milestone **Automation** Status Is Muted December 1.0 (2022) Manual Actual No

Pre-conditions

- 1. В аккаунте есть продукт №1 с матчами с примененный "Test" тегом и продукт с матчами №2 с примененным "DT test" динамическим тегом.
- 2. В "Gross margin" виджете в первой диаграмме выбрано в фильтрах "Test" тег, во второй диаграмме "DT test" динамический тег.
- 3. Перейти на Dashboard страницу.

Step	1
Action	В "Gross margin" виджете навести курсором на название первой диаграммы внизу.
nput data	3
Expected result	 В названии отображается: "Test". Отображается рассчитанное значение среднего марджина. При наведении курсором в тултипе отображается: "Tags: Test". В диаграмме отображается: 1 products.
Step	2
Action	В "Gross margin" виджете навести курсором на название второй диаграммы внизу.
Input data	3
Expected result	 В названии отображается: "DT test". Отображается рассчитанное значение среднего марджина. При наведении курсором в тултипе отображается: "Dynamic Tags: DT test". В диаграмме отображается: 1 products.
Step	3
Action	На Product list странице активировать чекбокс продукта №1 -> "Select action" -> "Tags" -> "Add tag".
Input data	3
Expected result	1. Ha Product list странице открылся "Add tag" попап, где отображается "Test" тег.
Step	4

Expected result	 Открылся "Delete tag" попап. После удаления тега "Delete tag" попап закрылся. Отображается "Tag deleted" сообщение.
Step	5
Action	Ha Dynamic tags странице удалить "DT test" динамический тег.
Input data	a
Expected result	1. Отображается "Dynamic tag "Test" deleted" сообщение.
Step	6
Action	Перейти на Dashboard страницу и навести курсором на название первой и второй диаграмм в "Gross m argin" виджете.
Input dat	a
Expected result	 В "Gross margin" виджете в первой и второй диаграммах отображается: No data Отображается рассчитанное значение среднего марджина: 0% on avg. В названии первой диаграммы отображается: "Test (removed)", при наведении на название в тултипе отображается: "Tags: Test (removed)". В названии второй диаграммы отображается: "DT test (removed)", при наведении на название в тултип е отображается: "Dynamic tags: DT test (removed)".
Step	7
Action	Нажать на кнопку редактирования первой диаграммы и нажать на "Reset to default" кнопку.
Input data	9
Expected result	1. В диаграмме отображается кол-во всех продуктов в аккаунте, в названии диаграммы отображается: " All products".
Step	8
Action	Нажать на кнопку редактирования второй диаграммы и нажать на "Reset to default" кнопку.

Expected 1. В диаграмме отображается кол-во всех продуктов в аккаунте, в названии диаграммы отображается: " result All products".

Input data

E2E: Dashboard -> Gross margin widget -> Reset to default button

Reset to default settings "Gross margin" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
December 1.0 (2022)	Manual	Actual	No

Pre-conditions

- 1. Account has products with attribute cost and products not with this attribute.
- 2. "Gross margin" widget is added and filtred by custom settings (Brands, Product types, Tags, Labels and own stock status and domains was choosen on 3 pie charts).
- 3. Dashboard page is opened.

Steps to reproduce

```
Step
          1
          Check result on database
Action
          SELECT * FROM `widgets` WHERE `account id` = <span style="color:var(--md-font-color-failed)">account id <
Input
          /span>and user_id = <span style="color:var(--md-font-color-failed)">user_id </span>AND id =<span style="c
data
          olor:var(--md-font-color-failed)"> id_your_grooss_margin_widget</span>
          In the fields params displayed JSON which displayed your filters:
          [
           "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
          "brandslds": [
          <span style="color:var(--md-font-color-failed)">brand_id</span>
          <span style="color:var(--md-font-color-failed)">label id</span>
          ],
          "domainsIds": [
          <span style="color:var(--md-font-color-failed)">domain id</span>
          "dynamicTags": [
          <span style="color:var(--md-font-color-failed)">dynamic_tag_id</span>
          "mainFilters": null,
          "stockStatus": [
          <span style="color:var(--md-font-color-failed)">"stock_status"</span>
          "customLabels": [
          <span style="color:var(--md-font-color-failed)">"name"</span>
          "customFilters": null,
          "productTypesIds": [
          <span style="color:var(--md-font-color-failed)">product type id</span>
          },
          {
          "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
          "brandsIds": [
           <span style="color:var(--md-font-color-failed)">brand id</span>
          "labelsIds": [
```

label id

domain_id

"domainsIds": [

],

```
Expected"dynamicTags": [
           <span style="color:var(--md-font-color-failed)">dynamic_tag_id</span>
result
           "mainFilters": null,
           "stockStatus": [
           <span style="color:var(--md-font-color-failed)">"stock status"</span>
           "customLabels": [
           <span style="color:var(--md-font-color-failed)">"name"</span>
           "customFilters": null,
           "productTypesIds": [
           <span style="color:var(--md-font-color-failed)">product_type_id</span>
           },
           "origin": <span style="color:var(--md-font-color-failed)">"origin"</span>,
           "brandslds": [
           <span style="color:var(--md-font-color-failed)">brand_id</span>
           "labelsIds": [
           <span style="color:var(--md-font-color-failed)">label id</span>
           "domainsIds": [
           <span style="color:var(--md-font-color-failed)">domain_id</span>
           "dynamicTags": [
           <span style="color:var(--md-font-color-failed)">dynamic_tag_id</span>
           "mainFilters": null,
           "stockStatus": [
           <span style="color:var(--md-font-color-failed)">"stock_status"</span>
           <span style="color:var(--md-font-color-failed)">"name"</span>
           "customFilters": null,
           "productTypesIds": [
           <span style="color:var(--md-font-color-failed)">product type id</span>
           }
          ]
          2
Step
          Click on the red pen "Edit" button inside the first pie chart on "Gross margin" widget
Action
          Click on "Reset to default" button
Input
data
Expected 1. "Widget was successfully updated" information message was displayed 2. The widget editing popup is closed.
result
          3. First pie chart It shows the gross margin for all products
Step
          3
          Click on the red pen "Edit" button inside the second pie chart on "Gross margin" widget
Action
          Click on "Reset to default" button
Input
data
Expected. "Widget was succession, a.g.
The widget editing popup is closed.
             "Widget was successfully updated" information message was displayed
          3. Second pie chart It shows the gross margin for all products
```

Click on the red pen "Edit" button inside the third pie chart on "Gross margin" widget Action Click on "Reset to default" button

Input data

Expected. "Widget was successfully updated" information message was displayed result. The widget editing popup is closed.

result

3. Third pie chart It shows the gross margin for all products

5 Step

Action Check result on database

Input data

```
In the fields params displayed JSON which displayed your filters:
            "origin": null,
            "brandslds": [],
            "labelsIds": [],
            "domainsIds": null,
            "dynamicTags": [],
            "mainFilters": null,
            "stockStatus": [],
            "customLabels": [],
            "customFilters": null,
            "productTypesIds": []
            },
           {
"origin": null,
            "brandslds": [],
            "labelsIds": [],
Expected "domainsIds": null, result "dynamicTags": [],
result
           "mainFilters": null,
            "stockStatus": [],
            "customLabels": [],
            "customFilters": null,
            "productTypesIds": []
           },
            "origin": null,
            "brandslds": [],
            "labelsIds": [],
            "domainsIds": null,
            "dynamicTags": [],
            "mainFilters": null,
            "stockStatus": [],
            "customLabels": [],
            "customFilters": null,
            "productTypesIds": []
            }
           j
```

Dashboard-> "Product price history" widget -> Calendar work

Period representation and a calendar work in the "Product price history" widget. By default widget shows dots for the last 7 days.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- There are products with matches and price changes history for a few months in the account.
 Dashboard page is opened.
- 3. "Product price history" widget is added.

Step	1
Action	Click on "FROM" and pick the date on the calendar, that is a month before today.
Input data	
Expected result	 Calendar popup shown. Popup closed. Near to the "From" displayed the date that was a month ago, near to the "To" displayed the actual date. The graph shows points from the day that was a month ago to the point that it shows today. Pay attention that the step between the points is one day.
Step	2
Action	Click on "From"and pick the date on the calendar, that is a 3 month before today.
Input data	
Expected result	 Calendar popup shown. Popup closed. Near to the "From" displayed the date that was 3 month ago, near to the "To" displayed the actual date. The graph shows points from the day that was 3 month ago to the point that it shows today.
Step	3
Action	Click on "to" and pick the date on the calendar that is a week before today.
Input data	
Expected result	 Calendar popup shown. Popup closed. Near to the "to" displayed the date that was a week ago, near to the "From" displayed the date that was 3 month. The graph shows points from the day that was 3 month ago to the point that was a week ago.
Step	4

Action	Reload the page.
Input data	
Expected result	 Page is updated. "Product price history" widget shows default period - 7 days before - today.

Dashboard -> "Product price history" widget -> "Show" dropdown->Select/Deselect All

SeverityPriorityBehaviorTypeO NormalO MediumPositiveUsabilityMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. There are products with matches and price changes history in the account.
- 2. Dashboard page is opened.
- 3. "Product price history" widget added.

Step	1
Action	Click on the "Show" button on the "Product price history" widget.
Input data	
Expected result	 Dropdown with domains and selected checkboxes was shown. Next to "[✓]Deselect all" is number of domains selected. The first after "[✓]Deselect all" is own domain.
Step	2
Action	Click on checkbox on the left to "Deselect all[/]".
Input data	
Expected result	 All checkboxes became unchecked. "[✓]Deselect all" changed to "[]Select all". The chart is not displayed. "No data to display" message instead is shown.
Step	3
Action	 Enter domains name into the search field. Chose domain.
Input data	test.dk
Expected result	 The domain name is displayed in the search field. The searched domain is displayed below. The domain checkbox is filled with a checkmark. The main checkbox is displayed as "[/]Deselect all". Lines of domain appeared on the chart.
Step	4
Action	 Clear the search field. Pay attention to domains checkboxes. Hover over a point on the chart.

Input data

Expected result

- 1. The search field has been cleared.
- 2. The main checkbox is displayed as "[-]Deselect all".
- 3. Selected domain checkbox is filled with a checkmark, no other domains selected.
- 4. The tooltip reflecting the date and price of the selected domain, stock status (if competitor has it), them c olors is shown.

Step

5

Action

On the "Show" dropdown click on checkbox on the left to "[-] Deselect all".

Input data

Expected result

- 1. All checkboxes became unchecked.
- 2. "[-] Deselect all" changed to "[]Select all".
- 3. The chart is not displayed.
- 4. "No data to display" message instead is shown.

6 Step

Action

On the "Show" dropdown click on checkbox on the left to "[]Select all".

Input data

Expected

result

- 1. All checkboxes became checked.
- 2. "Select all[]" changed to "Deselect all[✓]".
- 3. The chart is displayed with all selected domains and their price changes.

Dashboard -> "Product price history" widget -> "Show" dropdown-> Search

Severity **Priority Behavior Type** O Medium Usability O Normal Positive Milestone **Automation Status** Is Muted Manual Actual

Pre-conditions

Log in as a User (Admin,RA).

There are products with matches and price changes history in the account.

Dashboard page is opened.

"Product price history" widget added. Period- one week before today.

Product "A" has soft-deleted competitors from domain "test.com"-deleted 8 days ago; and soft-deleted competitors from domain "sd-domain.com"- deleted 2 days ago. Product "A" has "fashion.com", "davidson.dk" and "davidoff.dk" competitors.

Step	1
Action	"A" product is displayed on the "Product price history" widget. Click on the "Show" button.
Input data	
Expected result	 Dropdown with domains and selected checkboxes was shown. Next to "[/]Deselect all" is number of domains selected. The first after "[/]Deselect all" is own domain.
Step	2
Action	Click "Deselect all". On the search bar type domain "test.com".
Input data	test.com
Expected result	 "[/]Deselect all" changed to "[]Select all" There are no results for search. Soft-deleted competitors and domains are not displayed on the "Product price history" widget ("Show" dro pdown and graphs).
Step	3
Action	Clear search and type "sd-domain.com".
Input data	sd-domain.com
Expected result	 There are results for search for the period till competitor was soft-deleted (2 days ago). No graphs for this competitor for 2 last days.
Step	4
Action	Clear search and type "d"

Input data	d	
Expected result	"davidson.dk" and "davidoff.dk" competitors are displayed in the dropdown.	
Step	5	
Action	Enter "david"	
Input data	david	
Expected result	"davidson.dk" competitor is displayed in the dropdown.	

Dashboard -> "Product price history" widget -> Arrow switch

Arrow switching of products in the "Product price history" widget

SeverityPriorityBehaviorTypeO NormalO MediumPositiveUsabilityMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

Log in as a User (Admin, RA).

There are products with matches and price changes history in the account.

Dashboard page is opened.

Step	1
Action	Click on the red "Right arrow" on the "Product price history" widget and flip through the products
Input data	
Expected re	esult 10 different products was displayed. Pay attention to the 1st and 10th products
Step	2
Action	Click on the red "Left arrow" on the "Product price history" widget when the 1st product is displayed
Input data	
Expected re	esult The 10th product is displayed

[&]quot;Product price history" widget with 10 products is shown.

Dashboard -> "Product price history" widget -> Redirecting to the Product info page and back

Redirecting to the Product info page with selected product on the "product price history" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User (Admin, RA).

There are products with matches and price changes history in the account.

Dashboard page is opened.

Step	1
Action	Click on the product image below the widget's title.
Input data	
Expected result	The Product info page of the selected product was displayed.
Step	2
Action	Click on the "Back to dashboard" button.
Input data	
Expected result	The Dashboard page is opened.

[&]quot;Product price history" widget added.

Dashboard -> "Product price history" widget -> Redirecting to the url-link page

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User (Admin, RA).

There are products with matches and price changes history in the account.

Dashboard page is opened.

"Product price history" widget added.

Step	1
Action	Click on the product title below the product image on the "Product price history" widget.
Input data	
Expected result	The product page on the site opened in the background tab.

Dashboard-> "Product Price history" widget-> Soft deleted competitors

By default "Product Price history" widget shows dots for the last 7 days. Competitors (matches) are not displayed on the widget when they are soft-deleted.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

Pre-conditions

- 1. There are products with matches and price changes history for a few months in the account.

- Dashboard page is opened.
 "Product price history" widget is added.
 "Test" product has "A" competitor match for a few months.

Step	1
Action	Find "Test" product on the "Product Price history" widget.
Input data	
Expected result	 "Test" product displayed on the "Product Price history" widget. "A" competitor displayed with price changes history for few month till today.
Step	2
Action	In the DB set `deleted_at` for a "A" competitor - date week ago.
Input data	
Expected result	In the DB "A" competitor is soft deleted from the day week ago.
Step	3
Action	On the "Product Price history" click on "FROM" and pick the date on the calendar, that is a month before today.
Input data	
Expected result	 Near to the "From" displayed the date that was a month ago, near to the "To" displayed the actual date. The graph shows points from the day that was a month ago to the point that it shows today.
Step	4
Action	Pay attention on the "Test" product on the "Product Price history" widget. Find "A" competitor.
Input data	

Expected result

- "Test" product displayed on the "Product Price history" widget.
 "A" competitor is not displayed from the day week ago, price changes history before this date is displaye d.

Dashboard-> "Product Price history" widget-> Competitors as selected and displayed in the widget after changing the date

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть быть хотя бы 1 продукт с валидным значением "display price" и 5 конкурентами, которые были добавлены в аккаунт минимум неделю назад.
- 2. Открыть "Dashboard" страницу.
- 3. Добавить виджет "Product price history"

Step	1
Action	Раскрыть дропдаун "Show"
Input data	
Expected result	1. Отображается активный "Deselect all" [\checkmark] чекбокс 2. Отображаются активные чекбоксы своего домена и всех конкурентов.
Step	2
Action	Деактивировать чекбоксы 2-х доменов.
Input data	
Expected result	 Отображается неактивный "Deselect all" [-] чекбокс с минусом. Отображаются активные чекбоксы 3-х конкурентов и своего домена. Они же отображаются и на г рафике. Отображаются неактивные чекбоксы 2-х конкурентов.
Step	3
Action	Изменить дату (начала или конца) отображения информации в виджете с помощью календаря на 1 день.
Input data	
Expected result	 Диапазон дат изменен. На графике отображается информация о 3-х конкурентах и о своем домене. Конкуренты, которые были деактивированы не отображаются.
Step	4
Action	Раскрыть дропдаун "Show"
Input data	

Expected result

- Отображается неактивный "Deselect all" [-] чекбокс с минусом.
 Отображаются активные чекбоксы 3-х конкурентов и своего домена.
 Отображаются неактивные чекбоксы 2-х конкурентов.

Dashboard-> "Product Price history" widget-> Competitors as selected and displayed in the widget after changing the date -> No data

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть быть хотя бы 1 продукт созданный 7 дней назад с валидным значением "display price" и 5 конкурентами, которые были добавлены в аккаунт 7 дней назад.
- 2. Открыть "Dashboard" страницу.
- 3. Добавить виджет "Download price history"

Step	1
Action	Раскрыть дропдаун "Show"
Input data	
Expected result	 Отображается активный "Deselect all" [✓]чекбокс Отображаются активные чекбоксы своего домена и всех конкурентов.
Step	2
Action	Изменить дату (начала и конца) отображения информации в виджете с помощью календаря на 10 дней назад.
Input data	
Expected result	 Диапазон дат изменен. На графике отображается "No data to display" сообщение.
Step	3
Action	Раскрыть дропдаун "Show"
Input data	
Expected result	1. Отображается "no data" сообщение.

Dashboard-> "Product Price history" widget-> Chart -> hover over point (competitor changes)

Отображение информации по домену при наведении на точку на графике.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.

- Конкурент №1 имеет цену 150, availability Null.
 Конкурент № 2 имеет цену 220, availability in stock.
 Конкурент № 2 имеет цену 220, availability out of stock.
- 5. Свой домен имеет цену 200, availability in stock.
- 6. Открыть Dashboard страницу.
- 7. Добавить "Product price history" виджет.

Step	1
Action	Ha "Product price history" виджете навести ховер напротив 150, на точку домена конкурента №1, сег одняшняя дата.
Input data	
Expected result	Отображается тултип с сегодняшней датой, названием домена, ценой и прочерком для availability.
Step	2
Action	Ha "Product price history" виджете навести ховер напротив 220, на точку домена конкурента №2 и № 3, сегодняшняя дата.
Input data	
Expected result	Отображается тултип с сегодняшней датой, данные о домене. • домен №2, 220, in stock • домен №3, 220, out of stock
Step	3
Action	Ha "Product price history" виджете навести ховер напротив 200, на точку своего домена, сегодняшня я дата.
Input data	
Expected result	Отображается тултип с сегодняшней датой, названием домена, ценой и availability - in stock

Dashboard-> "Product Price history" widget-> Chart -> hover over date (competitor changes)

Отображение информации на виджете при наведении на дату.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.
- 2. У товара №1 цена 345, есть 4 матчей:

конкурент "А" - имеет цену 342, out of stock; конкурент "В" - имеет цену 609, in stock;

конкурент "С" - имеет цену 345, "availabillity" - NULL; конкурент "D" - имеет цену 345, был МУ с 21.0623 по 24.06.23;

- 3. Открыть Dashboard.
- 4. Добавить "Product Price history" виджет.

Step	1
Action	На "Product price history" виджете навести ховер на дату 23.06.23.
Input data	
	Появилось окно, где отображается:
Expected result	 дата 2023-06-23, список доменов - свой домен первый, далее в алфавитном порядке, цена и "availability" домен конкурента"D" отображается серым с задизейбленным чекбоксом.
Step	2
Action	Снять выбор с чекбокса домена "В".
Input data	
Expected result	 Чекбокс снят. График перестроился. Точка для домена "В" не отображается за дату 23.06.23.
Step	3
Action	Перезагрузить страницу. Обратить внимание на график для продукта №1 на "Product price history" виджете.
Input data	
Expected result	 Страница перегрузилась. Точка для домена "В" не отображается за дату 23.06.23.

Action

На "Product price history" виджете навести ховер на дату 23.06.23.

Input data

Появилось окно, где отображается:

- 1. Дата 2023-06-23. Expected result 2. Список доменов свой домен первый, далее в алфавитном порядке, цена и "availability". 3. В списке "В" домен отображается предпоследним с пустым чекбоксом. 4. Домен конкурента"D" отображается последним, серым с задизейбленным чекбоксом.

Dashboard-> "Product Price history" widget-> Chart -> Lock domain selection

Запоминание выборки доменов на "Product Price history" для определенного продукта.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.
- 2. У продукта №1 есть 16 матчей.
- 3. У продукта №2 есть 11 матчей.
- 4. "Product Price history" виджет добавлен на Dashboard.
- 5. Открыть Dashboard.

Step	1
Action	Ha "Product Price history" виджете найти продукт №1. Раскрыть "Show" дропдаун.
Input data	
Expected result	 Продукт №1 отображается на виджете, с графиком истории цен своего домена и конкурентов в пер иод сегодня-неделя назад. В "Show" дропдауне отображается список со всеми доменами-матчами, первый в списке- свой домен . Все домены выбраны, отображается "Deselect all" с заполненной галочкой, 16 selected. "Lock domain selection" тугл выключен.
Step	2
Action	Снять выбор в чекобке "Deselect all".
Input data	
Expected result	 Все домены не выбраны. График пуст "No data to display".
Step	3
Action	Выбрать 5 любых доменов и включить опцию "Lock domain selection"
Input data	
Expected result	 В дропдауне выбрано 5 доменов. Отображается "5 selected". "Lock domain selection" тугл включен. На графике отображаются линии истории цен выбранных 5 доменов в период сегодня-неделя назад.

Action	Перезагрузить страницу. Обратить внимание на продукт №1 на "Product Price history" виджете.
Input data	
Expected result	Страница перезагружена. На графике отображаются линии истории цен выбранных 5 доменов в период сегодня-неделя назад.
Step	5
Action	Раскрыть "Show" дропдаун.
Input data	
Expected result	 В дропдауне выбрано 5 доменов. Отображается "5 selected". "Lock domain selection" тугл включен.
Step	6
Action	Ha "Product Price history" виджете перелиснуть на продукт№2. Раскрыть "Show" дропдаун. Обратить внимание на выборку доменов.
Input data	
Expected	 Продукт №2 отображается на виджете, с графиком истории цен своего домена и конкурентов в пер иод сегодня-неделя назад. В "Show" дропдауне отображается список со всеми доменами-матчами, первый в списке- свой домен

3. Все домены выбраны, отображается "Deselect all" с заполненной галочкой, 1 selected. 4. "Lock domain selection" тугл выключен.

Expected result

Dashboard-> "Product Price history" widget-> Chart -> Lock domain correlation

Взаимосвязь "Lock domain" выборки для конкретного продукта на виджете на Дашборде и на Продукт инфо. Смена выборки происходит автоматически при включенном "Lock domain selection".

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.
- 2. У продукта №1 есть 16 матчей.
- 3. "Product Price history" виджет добавлен на Dashboard.
- 4. Ha Dashboard странице на "Product Price history" виджете для продукта №1 включен "Lock domain selection" для 5 доменов.
- 5. Открыть Product info продукта №1 включен.

Step	1
Action	Ha Product info продукта №1 , "Product Price history" виджете раскрыть "Show" дропдаун.
Input data	
Expected result	 Продукт №1 отображается на виджете, с графиком истории цен 5-ти доменов в период сегодня-не деля назад. В "Show" дропдауне выбрано 5 доменов. Отображается "5 selected". "Lock domain selection" тугл включен.
Step	2
Action	Нажать на чекбоксы 2-х не выбранных доменов.
Input data	
Expected result	 Продукт №1 отображается на виджете, с графиком истории цен 7-ми доменов в период сегодня-не деля назад. В "Show" дропдауне выбрано 7 доменов. Отображается "7 selected". "Lock domain selection" тугл включен.
Step	3
Action	Перезагрузить Product info страницу.
Input data	
Expected result	Product info страница перезагружена. На графике отображаются линии истории цен выбранных 7 доменов в период сегодня-неделя назад.
Step	4

Action	Раскрыть "Show" дропдаун.
Input data	
Expected result	 В дропдауне выбрано 7 доменов. Отображается "7 selected". "Lock domain selection" тугл включен.
Step	5
Action	Перейти на Dashboard. На "Product Price history" виджете найти продукт №1. Раскрыть "Show" дропдаун.
Input data	
Expected result	 Продукт №1 отображается на виджете, с графиком истории цен 7-ми доменов в период сегодня-не деля назад. В "Show" дропдауне выбрано 7 доменов. Отображается "7 selected". "Lock domain selection" тугл включен.
Step	6
Action	Снять выбор со всех доменов.
Input data	
Expected result	 Все домены не выбраны. В дропдауне отображается "0 selected". График пуст "No data to display". "Lock domain selection" тугл включен.
Step	7
Action	Перейти на Product info продукта №1 , "Product Price history" виджете раскрыть "Show" дропдаун.
Input data	
Expected result	1. Все домены не выбраны. 2. В дропдауне отображается "0 selected".

Dashboard-> "Product Price history" widget-> Chart -> Hover over date

Отображение тултипа при наведении курсора на дату. В списке доменов:

- Сначала отображаются домены, что выбраны на виджете.
- Потом домены, что анчекнуты по дате.
- Потом домены, что не отображают истории по данной дате (задизейблены).

Severity	Priority	Behavior	Туре
O Normal	↓ Low	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

- 1. В аккаунте есть товары с матчами с историей изменения цен за неделю.
- 2. На Dashboard странице добавлен "Product price history" виджет.
- 3. Открыть Dashboard страницу.

Step	1
Action	Ha "Product price history" виджете навести ховер на дату.
Input data	
Expected result	 Виджет дефолтно отображает период времени в 8 дней - "Сегодня" - "Неделя назад". Отобразилось окно со списком доменов за дату уууу-mm-dd. Домены отмечены чекбоксами разных цветов, соответствуют цвету на графике. Напротив домена отображается цена матча (с валютой если вкл и есть), сток статус.
Step	2
Action	В окне за дату анчекнуть домен.
Input data	test.dk
Expected result	 Галочка снята с домена, домен отображается серым. Домен не поменял свою позицию в списке. Напротив данного домена не отображается цена и сток статус. На виджете у домена пропала точка за данную дату, график стал прерывистым.
Step	3
Action	Навести ховер на выбранную дату еще раз.
Input data	
Expected result	1. Домен отображается в конце списка, неактивным, но доступным для выбора, без цены и сток статуса.

Step	4
Action	Нажать на "Show" дропдаун и анчекнуть собственный домен.
Input data	
Expected result	 Собственный домен неактивный в дропдауне. Собственный домен не отображается на виджете.
Step	5
Action	Навести ховер на выбранную дату еще раз.
Input data	
Expected result	1. В конце списка отображается собственный домен, задизейбленным, недоступным для выбора. 2. Предпоследним отображается домен test.dk.

Dashboard-> "Product Price history" widget-> Chart -> Lock domain -> Change period

При смене периода на виджете выборка "Lock domain" должна сохраняться.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.

- У продукта №1 есть 16 матчей.
 "Product Price history" виджет добавлен на Dashboard.
 На "Product Price history" виджете для продукта №1 включен "Lock domain selection" для 5 доменов.
- 5. Открыть Dashboard.

Step	1
Action	Ha "Product Price history" виджете найти продукт №1. Раскрыть "Show" дропдаун.
Input data	
Expected result	 Продукт №1 отображается на виджете, с графиком истории цен 5-ти доменов в период сегодня-не деля назад. В "Show" дропдауне выбрано 5 доменов. Отображается "5 selected". "Lock domain selection" тугл включен.
Step	2
Action	Сменить период отображения на виджете на месяц назад- 2 дня назад.
Input data	
Expected result	 Период изменился. В "From" - дата месяц назад, в "То"- 2 дня назад. На графике отображаются выбранные 5 доменов с историей цен с момента появления матчей. Если матч был МУ- линия прерывается.
Step	3
Action	Раскрыть "Show" дропдаун.
Input data	
Expected result	 B "Show" дропдауне выбрано 5 доменов. Отображается "5 selected". "Lock domain selection" тугл включен.

Dashboard-> "Product Price history" widget-> Chart -> hover over point (own changes)

Отображение информации по домену при наведении на точку на графике.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
September 1.0	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.
- 2. Есть история изменения цен и сток статуса своего продукта за позавчера, вчера и сегодня.
 - позавчера:
 - -- цена 150, availability Null;
 - -- цена 220, availability in stock;
 - -- цена 100, availability preorder.
- 3. Перейти на Dashboard страницу.
- 4. Product price history виджет добавлен.

Step	1
Action	1. На "Product price history" виджете навести ховер на график за позавчера.
Input data	
Expected result	1. Отображается тултип с позавчерашней датой, названием аккаунта, ценой 150 и availability с прочер ком.
Step	2
Action	1. На "Product price history" виджете навести ховер на график за вчера.
Input data	
Expected result	1. Отображается тултип с вчерашней датой, названием аккаунта, ценой 220 и availability in stock (с зел еным кружочком).
Step	3
Action	1. На "Product price history" виджете навести ховер на график за сегодня.
Input data	
Expected result	1. Отображается тултип с сегодняшней датой, названием аккаунта, ценой 100 и availability preorder (с желтым кружочком).

Dashboard-> "Product Price history" widget-> Chart -> hover over date (own changes)

Отображение информации на виджете при наведении на дату.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Usability
Milestone	Automation	Status	Is Muted
September 1.0	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть товары с матчами, что были добавлены минимум неделю назад.
- 2. Есть история изменения цен и сток статуса своего продукта за позавчера, вчера и сегодня.
 - позавчера:
 - -- цена 150, availability Null;
 - -- цена 220, availability in stock;
 - -- цена 100, availability preorder.
- 3. Перейти на Dashboard страницу.
- 4. Product price history виджет добавлен.

Step	1
Action	1. На "Product price history" виджете навести ховер на позавчерашнюю дату.
Input data	
Expected result	Появилось окно, где отображается: 1. позавчерашняя дата, 2. список доменов - свой домен первый, далее в алфавитном порядке, цена своего продукта 150 и availa bility с прочерком.
Step	2
Action	1. На "Product price history" виджете навести ховер на вчерашнюю дату.
Input data	
Expected result	Появилось окно, где отображается: 1. вчерашняя дата, 2. список доменов - свой домен первый, далее в алфавитном порядке, цена своего продукта 220 и availa bility in stock (с зеленым кружочком).
Step	3
Action	1. На "Product price history" виджете навести ховер на сегодняшнюю дату.
Input data	1

Появилось окно, где отображается:

Expected result

- сегодняшняя дата,
 список доменов свой домен первый, далее в алфавитном порядке, цена своего продукта 100 и availa bility preorder (с желтым кружочком).

E2E: Dashboard -> "Product price history" widget -> Deleting a product

Deleting a product on the "Product price history" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User (Admin, RA).

There are products with matches and price changes history in the account.

Dashboard page is opened.

"Product price history" widget was added.

1
Choose and copy product title below the product image on the "Product price history" widget. Delete product from widget in the editing mode.
 "CHOOSE PRODUCTS FOR PRICE HISTORY" pop-up with 10 product items displayed. The title of the selected product has disappeared from the list. Editin pop-up was closed. "Widget was successfully updated" information massage displayed.
2
Click on the red "Right arrow" flip through the products on the "Product price history" widget.
The selected product is not on the widget. Pay attention to the number of displayed products (9)

E2E: Dashboard -> "Product price history" widget -> Adding a product

Adding a new product on the "Product price history" widget

2

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

1

- 1. There are products with matches and price change history in the account.
- 2. The Dashboard page is opened.
- 3. One of the 10 products on the "Product price history" widget is deleted.

Step	1
Action	 Add the "Test" product to the "Product price history" widget that is in the product list, but not on the widg et. In the editing mode, input field, start typing the title "Test".
Input data	
Expected result	 "CHOOSE PRODUCTS FOR PRICE HISTORY" pop-up with 9 product items displayed. The list of goods with unchecked marks that start with "Test" is shown. The title is displayed in the input field. The product is checked on the list of the products below the input field and the product title is displayed as 10th product on the right. Editing pop-up closed. "Widget was successfully updated" message is displayed.
Step	2
Action	1. Click on the red "Left arrow" and flip through the products.
Input data	
Expected result	1. The added "Test" product is on the widget. Pay attention to the number of displayed products (10).

E2E: Dashboard-> Product price history widget-> Check products limit in widget

Проверить лимит продуктов в виджете.

По дефолту при создание ввиджете добавляется 10 продуктов. Максимальное количесто которое можно добавить в один виджет 15 продуктов

Severity	Priority	Behavior	Туре
^ Major	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	2	3	3

Pre-conditions

- 1. Открыта страница Dashboard
- 2. В аккаунте минимум 16 продуктов с матчами
- 3. Добавленный виджет Product price history с 10 продуктами

Step	1
Action	Открыть окно редактирования виджета Product price history
Input data	
Expected result	 Открылось окно EDIT PRODUCT PRICE HISTORY WIDGET Под полем поиска отображаются выбранные продукты 10 штук
Step	2
Action	Нажать на поле поиска ввести часть имени продуктов
Input data	Достаточно одной буквы что бы появился список
Expected result	1. Открылся список продуктов - вверхней части отображаются выбранные продукты с активными чек- боксами
Step	3
Action	Добавить 5 продуктов
Input data	
Expected result	1. После выбора 15-го продукта не выбранные продукты стали серого цвета и чекбоксы возле них не кликабельные
Step	4
Action	Сохранить изменения
Input data	

Expected result	1. Окно EDIT PRODUCT PRICE HISTORY WIDGET закрылось 2. Отобразилось сообшение "Widget was successfully updated"
Step	5
Action	Повторно открыть окно редактирования виджета Product price history
Input data	
Expected result	1. Открылось окно EDIT PRODUCT PRICE HISTORY WIDGET 2. Под полем поиска отображаются выбранные продукты 15 штук

Dashboard -> Competitor Activity widget -> Switching radiobuttons

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
_	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- У конкурента "test.dk" сегодня изменялась цена.
 У конкурента "babor.dk" сегодня изменялся сток статус.
- 4. Открыть страницу "Dashboard"

Step	1
Action	Добавить виджет Competitor Activity" на страницу "Dashboard".
Input data	
Expected result	 Виджет "Competitor Activity" добавлен. Отображается в состоянии по умолчанию, отображая 5 конкурентов, включая "test.dk" с последними изменениями цен в аккаунте. Радиобаттон "Price" - активный.
Step	2
Action	Нажать на радиобаттон "Stock status".
Input data	
Expected result	 Радиобаттон "Price" - стал неактивный, а "Stock status" - активный. В виджете отображается 5 конкурентов с последними изменениями сток статуса, включая "babor.dk" в аккаунте.
Step	3
Action	Обновить страницу.
Input data	
Expected result	 Страница обновилась. В виджете отображается 5 конкурентов, включая "test.dk" с последними изменениями цен в аккаунт е. Радиобаттон "Price" - активный, "Stock status" - неактивный.

Dashboard -> Competitor Activity widget -> Competitor activity page-> Price radiobutton -> Search

SeverityPriorityBehaviorTypeO NormalO MediumPositiveFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкурент, у которого:
 - У конкурента "test.dk" сегодня изменялась цена, его gtin 886798030012, title "CamelBak Forge".
- 3. На странице "Dashboard" добавлен виджет "Competitor Activity".
- 4. Открыть страницу "Dashboard"

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input data	
Expected result	1. Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price" 2. Отображается список конкурентов, у которых изменялась цена за последние 24 часа.
Step	2
Action	Ввести в поиск значение
Input data	C C
Expected result	1. Отображаются все конкуренты.
Step	3
Action	Ввести в поиск значение
Input data	a Cam
Expected result	1. Отображаются все конкуренты, название которых начинается с "Cam"
Step	4
Action	Ввести в поиск значение
Input data	CamelBak Forge

Step	5
Action	Очистить поле поиска
Input data	
Expected result	1. Отображаются все конкуренты.
Step	6
Action	Ввести в поиск значение
Input data	8
Expected result	1. Отображаются все конкуренты.
Step	7
Action	Ввести в поиск значение
Input data	886
Expected result	1. Отображаются все конкуренты, gtin которых начинается с " 886"
Step	8
Action	Ввести в поиск значение
Input data	886798030012

Dashboard -> Competitor Activity widget -> Competitor activity page-> Stock status radiobutton -> Search

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкурент, у которого:
 - У конкурента "babor.dk" сегодня изменялся сток статус, его gtin 7318731769428, title "Lundhags Gneik"
- 3. На странице "Dashboard" добавлен виджет "Competitor Activity".
- 4. Открыть страницу "Dashboard"

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input dat	a
Expected result	1. Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price" 2. Отображается список конкурентов, у которых изменялась цена за последние 24 часа.
Step	2
Action	Нажать на радиобаттон "Stock status"
Input dat	9
Expected result	1. Отображается список конкурентов, у которых изменялся сток статус за последние 24 часа.
Step	3
Action	Ввести в поиск значение
Input dat	a L
Expected result	1. Отображаются все конкуренты.
Step	4
Action	Ввести в поиск значение
Input dat	a Lun
Expected result	1. Отображаются все конкуренты, название которых начинается с "Lun"

Step	5
Action	Ввести в поиск значение
Input data	a Lundhags Gneik
Expected result	1. Отображается конкурент "babor.dk" и другие, если у них название также включает в себя "Lundhags G neik" и изменялся сток статус за последние 24 часа.
Step	6
Action	Очистить поле поиска
Input data	a
Expected result	1. Отображаются все конкуренты.
Step	7
Action	Ввести в поиск значение
Input data	a 73
Expected result	1. Отображаются все конкуренты.
Step	8
Action	Ввести в поиск значение
Input data	a 731
Expected result	1. Отображаются все конкуренты, gtin которых начинается с " 731"
Step	9
Action	Ввести в поиск значение
Input data	a 7318731769428
Expected result	1. Отображается конкурент "babor.dk" и другие, если у них такой же gtin и изменялся сток статус за пос ледние 24 часа.

Dashboard -> Competitor Activity widget -> Competitor activity page-> Price radiobutton -> Dropdown

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкуренты, у которых:
 - У конкурента "test.dk" сегодня изменялась цена.
 - У конкурента "dakoo.dk" 10 дней назад изменялась цена.
 - У конкурента "bike.dk" 2 месяца назад изменялась цена.
- 3. Виджет "Competitor Activity" добавлен на страницу "Dashboard".
- 4. Открыть страницу "Dashboard".

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input data	
Expected result	 Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price" В дропдауне отображается "Last 24 hours" значение. Отображается список конкурентов, у которых изменялась цена за последние 24 часа, включая доме н "test.dk". Домены "dakoo.dk" и "bike.dk" не отображаются.
Step	2
Action	Выбрать в дропдауне "Last month" значение.
Input data	
Expected result	 В дропдауне отображается "Last month" значение. Отображается список конкурентов, у которых изменялась цена за последний месяц, включая домен "test.dk" и "dakoo.dk". Домен "bike.dk" не отображается.
Step	3
Action	Выбрать в дропдауне "Last 24 hours" значение.
Input data	
Expected result	 В дропдауне отображается "Last 24 hours" значение. Отображается список конкурентов, у которых изменялась цена за последние 24 часа, включая доме н "test.dk". Домены "dakoo.dk" и "bike.dk" не отображаются.

Dashboard -> Competitor Activity widget -> Competitor activity page-> Stock status radiobutton -> Dropdown

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкуренты, у которых:
 - У конкурента "test.dk" сегодня изменялся сток статус.
 - У конкурента "dakoo.dk" 10 дней назад изменялся сток статус.
 - У конкурента "bike.dk" 2 месяца назад изменялся сток статус.
- 3. Виджет "Competitor Activity" добавлен на страницу "Dashboard".
- 4. Открыть страницу "Dashboard".

•	
Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input data	
Expected result	 Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price". В дропдауне отображается "Last 24 hours" значение.
Step	2
Action	Нажать на радиобаттон "Stock status".
Input data	
Expected result	 Радиобаттон "Price" стал неактивным, а "Stock status" активным. В дропдауне отображается "Last 24 hours" значение. Отображается список конкурентов, у которых изменялся сток статус за последние 24 часа, включая "test.dk". Конкуренты "dakoo.dk" и "bike.dk" не отображаются.
Step	3
Action	Выбрать в дропдауне "Last month" значение.
Input data	
Expected result	 В дропдауне отображается "Last month" значение. Отображается список конкурентов, у которых изменялся сток статус за последний месяц, включая д омен "test.dk" и "dakoo.dk". Домен "bike.dk" не отображается.
Step	4
Action	Выбрать в дропдауне "Last 24 hours" значение.

Input data

Expected result

- В дропдауне отображается "Last 24 hours" значение.
 Отображается список конкурентов, у которых изменялся сток статус за последние 24 часа, включая домен "test.dk".
 Домены "dakoo.dk" и "bike.dk" не отображаются.

Dashboard -> Competitor Activity widget -> Redirecting to the Product info page and back

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкурент, у которого:
 - У конкурента "test.dk" сегодня изменялась цена, его gtin 886798030012, title "CamelBak Forge".
 - У конкурента "babor.dk" сегодня изменялся сток статус, его gtin 7318731769428, title "Lundhags Gneik".
- 3. На странице "Dashboard" добавлен виджет "Competitor Activity".
- 4. Открыть страницу "Dashboard".

Step	1
Action	Нажать на "Title" конкурента "test.dk".
Input data	
Expected result	 Произошло перенаправление на страницу "Product info" продукта №1. Цена, а также другая информация про конкурента "test.dk" на странице "Product info" совпадает с ин формацией на виджете.
Step	2
Action	Нажать на "Back to Dashboard" кнопку.
Input data	
Expected result	1. Открылась страница "Dashboard".
Step	3
Action	Нажать на радиобаттон "Stock status" на виджете "Competitor Activity".
Input data	
Expected result	1. Радиобаттон "Stock status" стал активным, а "Price" неактивным. 2. Отображается "babor.dk" конкурент.
Step	4
Action	Нажать на "Title" конкурента "babor.dk".
Input data	

1. Произошло перенаправление на страницу "Product info" продукта №1. Expected 2. Сток статус, а также другая информация про конкурента "babor.dk" на странице "Product info" совпад result ает с информацией на виджете. Step 5 Action Нажать на "Back to Dashboard" кнопку. Input data Открылась страница "Dashboard". Радиобаттон "Price" стал активным, а "Stock status" неактивным. Expected

3. В виджете отображается домен "test.dk".

result

Dashboard -> Competitor Activity widget -> Competitor activity page-> Redirecting to the Product info page

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У продукта №1 есть конкурент, у которого:
 - У конкурента "test.dk" сегодня изменялась цена, его gtin 886798030012, title "CamelBak Forge".
 - ∘ У конкурента "babor.dk" сегодня изменялся сток статус, его gtin 7318731769428, title "Lundhags Gneik".
- 3. На странице "Dashboard" добавлен виджет "Competitor Activity".
- 4. Открыть страницу "Dashboard".

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input data	
Expected result	 Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price". Отображается список конкурентов, у которых изменялась цена за последние 24 часа, включая домен "test.dk".
Step	2
Action	Нажать на "Title" конкурента "test.dk".
Input data	
Expected result	 Произошло перенаправление на страницу "Product info" продукта №1. Цена, а также другая информация про конкурента "test.dk" на странице "Product info" совпадает с ин формацией на виджете.
Step	3
Action	Нажать на "Back to competitor activity" кнопку.
Input data	
Expected result	1. Открылась страница "COMPETITOR ACTIVITY".
Step	4
Action	Нажать на радиобаттон "Stock status" на виджете "Competitor Activity".
Input data	

Expected result	1. Отображается список конкурентов, у которых изменялся сток статус за последние 24 часа, включая домен "babor.dk".
Step	5
Action	Нажать на "Title" конкурента "babor.dk".
Input data	
Expected result	 Произошло перенаправление на страницу "Product info" продукта №1. Сток статус, а также другая информация про конкурента "babor.dk" на странице "Product info" совпад ает с информацией на виджете.
Step	6
Action	Нажать на "Back to competitor activity" кнопку.
Input data	
Expected result	 Открылась страница "COMPETITOR ACTIVITY". Радиобаттон "Stock status" остался активным. Отображается список конкурентов, у которых изменялся сток статус за последние 24 часа, включая домен "test.dk".

Dashboard -> Competitor Activity widget -> Competitor activity page-> Price radiobutton-> Pagination

SeverityPriorityBehaviorTypeO NormalO MediumPositiveFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. За сегодня у 52 домена менялась цена.
- 3. В "Settings"-> User setup-> установлено 10 products per page.
- 4. На странице "Dashboard" добавлен виджет "Competitor Activity".
- 5. Открыть страницу "Dashboard".

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input data	3
Expected result	 Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price". Отображается 10 конкурентов, у которых изменялась цена за последние 24 часа. Порядок отображени я- сортировка по дате поступления. Отображается 1 - 10 out of 52 пагинация.
Step	2
Action	Перейти на "2" страницу.
Input data	3
Expected result	 Отображается следующие 10 конкурентов, у которых изменялась цена за последние 24 часа. Порядок отображения- сортировка по дате поступления. Отображается 10 - 20 out of 52 пагинация.
Step	3
Action	Перейти на "6" страницу.
Input data	3
Expected result	 Отображается 2 конкурента, у которых изменялась цена за последние 24 часа. Порядок отображения- сортировка по дате поступления. Отображается 50 - 52 out of 52 пагинация.
Step	4
Action	Нажать на ">" кнопку.

Input dat	a
Action	Нажать на "<" кнопку.
Step	5
result	1. Кнопка не кликабельна, ничего не произошло.

- Expected result 1. Отображается 10 конкурентов, у которых изменялась цена за последние 24 часа. Порядок отображени я- сортировка по дате поступления. 2. Отображается 40 50 out of 52 пагинация.

Dashboard -> Competitor Activity widget -> Competitor activity page-> Stock status radiobutton -> Pagination

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. За сегодня у 22 доменов менялся сток статус.
- B "Settings"-> User setup-> установлено 5 products per page.
 Ha странице "Dashboard" добавлен виджет "Competitor Activity".
- 5. Открыть страницу "Dashboard".

Step	1
Action	Нажать на кнопку "See all competitor activity" на виджете "Competitor Activity".
Input dat	a
Expected result	1. Открылась страница "COMPETITOR ACTIVITY" с активным радиобаттоном "Price".
Step	2
Action	Нажать на радиобаттон "Stock status".
Input dat	a
Expected result	 Отображается 5 конкурентов, у которых изменялся сток статус за последние 24 часа. Порядок отображения- сортировка по дате поступления. Отображается 1 - 5 out of 22 пагинация.
Step	3
Action	Перейти на "2" страницу.
Input dat	ra
	а 1. Отображается следующие 10 конкурентов, у которых изменялся сток статус за последние 24 часа. Пор ядок отображения- сортировка по дате поступления. 2. Отображается 5 - 10 out of 22 пагинация.
Expected	1. Отображается следующие 10 конкурентов, у которых изменялся сток статус за последние 24 часа. Пор ядок отображения- сортировка по дате поступления.
Expected result	 Отображается следующие 10 конкурентов, у которых изменялся сток статус за последние 24 часа. Пор ядок отображения- сортировка по дате поступления. Отображается 5 - 10 out of 22 пагинация.

Expected result	 Отображается 2 конкурента, у которых изменялся сток статус за последние 24 часа. Порядок отображ ения- сортировка по дате поступления. Отображается 20 - 22 out of 22 пагинация.
Step	5
Action	Нажать на ">" кнопку.
Input dat	a
Expected result	1. Кнопка не кликабельна, ничего не произошло.
Step	6
Action	Нажать на "<" кнопку.
Input dat	a

Expected result 1. Отображается 5 конкурентов, у которых изменялся сток статус за последние 24 часа. Порядок отобра жения- сортировка по дате поступления. 2. Отображается 15 - 20 out of 22 пагинация.

Dashboard -> Competitor Activity widget -> Filters, "Clear filters" button and change time period

Тест-кейс проверяет поботу фильтров а именно: Фильтрацию, кнопку "Clear filters" и что бы она не сбрасывалась при смене периода отображения времени

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У некоторых конкурентов ранее изменялся сток статус и цена.
- 3. Открыть страницу {Host}/product-list/competitor-activity
- 4. Период отображения времени установленный на Last 24 hours

Step	1
Action	Нажать на кнопку "Filters"
Input data	
Expected result	 Открылася интерфейс фильтров, справа отображатся такие табы: Save filters, Brands, Product types, Tags, Custom labels, Price strategy, Price position, Gross margin, Stock s tatus, Favorite, Dynamic Tags. Отображается кнопки "Save", "Closed" и задизеблинная кнопка "Save as template"
Step	2
Action	Выбрать любой фильтер
Input data	
Expected result	1. Чек-бокс фильтра стал активным
Step	3
Action	Выйти из интерфейса фильтров
Input data	
Expected result	 Интерфейс фильтров закрылись. Отображается "Clear filters" кнопка. В развернутом варианте виджета отображается отфильтрованная выборка
Step	4
Action	Выбрать период времени
Input data	Last month

Expected result	 Кнопка "Clear filters" осталась активной Период времени изменился с Last 24 hours на Last month Выборка изменилась
Step	5
Action	Нажать на "Clear filters" кнопку.
Input data	
Expected result	1. "Clear filters" кнопка исчезла.
Step	6
Action	Нажать на "Filters" кнопку.
Input data	

Expected result

 Открылася интерфейс фильтров, справа отображатся такие табы: Save filters, Brands, Product types, Tags, Custom labels, Price strategy, Price position, Gross margin, Stock s tatus, Favorite, Dynamic Tags.
 Отображается кнопки "Save", "Closed" и задизеблинная кнопка "Save as template"

Dashboard -> Competitor Activity widget -> Competitor activity page-> Check data in widget

Сравнить данные в виджете с базой данных. Написать запрос по которым можно проверить коректность отображенния данных в виджете

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. В настройках выбрано отображать "25 items per page".
- 3. За период в последние 30 дней были изменения цен и статуса доступности конкурентов.
- 4. На Dashboard странице добавлен "Competitor Activity" виджет, активен "Price" радиобаттон.
- 5. Открыть Dashboard страницу.

Steps to reproduce

C+0	_	1
Stei)	

Action

- 1. Обратить внимание на данные в "Competitor Activity" виджете.
- 2. Выполнить запрос в БД.

select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domain`, `competit tor_price_history`.`clear_new_price` as `new_competitor_price`,ROUND((100 * (CALC_PRICE_WITH_QUANTITY(s phinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.cle ar_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competit ors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference`, `competitors`.`availability` from `competitor_price_history` inner join competit ors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitors`.`id` inner join matchings on `matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_product s`.`id` where competitor_price_history.account_id = ACCOUNT_ID

Input data

Expected. Отображаются последние 5 изменений цен конкурентов. result 2. Значения на странице аналогичны выборке базе данных.

Step 2

Action Активировать радиобаттон "Stock status"

Input data

 $\mathsf{Expected}_{\mathsf{Paq}\mathsf{uo}\mathsf{battoh}}$ "Stock status" стал активен, а радиобаттон "Price" стал неактивен. result

Step 3

•

Action 1. Обратить внимание на данные в "Competitor Activity" виджете.

2. Выполнить запрос в БД.

Input data

select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products`.`availability `as `own_stock_status`, `domains`.`domain`, `competitors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availability_history`.`created_at` from `competitor_availability_history` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`com petitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where comp etitor_availability_history.account_id = ACCOUNT_ID a nd `competitor_availability_history`.`availability` is not null order by `competitor_availability_history`.`created_at` desc limit 5

Expected. Отображаются последние 5 изменений статуса доступности конкурентов.

result 2. Значения на странице аналогичны выборке базе данных.

Step 4

Action

- 1. Нажать на "See all competitor activity".
- 2. Выполнить запрос в БД.

Input data select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products`.`availability `as `own_stock_status`, `domains`.`domain`, `competitors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availability_history`.`created_at` from `competitor_availability_history` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.account_id = ACCOUNT_ID and `competitor_availability_history`.`created_at` >= 'ACCOUNT_ID and `competitor_availability_history`.`created_at` desc limit 25

Expected1. Отображаются последние 25 изменений статуса доступности конкурентов за последние 24 часа.

result 2. Значения на странице аналогичны выборке базе данных.

Step 5

Action

- 1. Выбрать период "Last month".
- 2. Выполнить запрос в БД.

Input data select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products`.`availability `as `own_stock_status`, `domains`.`domain`, `competitors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availability_history`.`created_at` from `competitor_availability_history` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.account_id = ACCOUNT_ID and `competitor_availability_history`.`created_at` >= 'ACCOUNT_ID and `competitor_availability_history`.`created_at` >= 'ACCOUNT_ID and `competitor_availability_history`.`created_at` availability_history`.`created_at` desc limit 25

Expected1. Отображаются последние 25 изменений статуса доступности конкурентов за последний 1 месяц.

result 2. Значения на странице аналогичны выборке базе данных.

Step 6

Action Активировать радиобаттон "Price"

Input data

Expected Paguoбatton "Stock status" стал активен, а радиобatton "Price" стал неактивен. result

Input

data

Action 1. Выполнить запрос в БД.

select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domain`, `competi tor_price_history`.`clear_new_price` as `new_competitor_price`,ROUND((100 * (CALC_PRICE_WITH_QUANTITY(s phinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.cle ar_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competit ors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference`, `competitors`.`availability` from `competitor_price_history` inner join competit ors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitors`.`id` inner join matchings on `matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_product s`.`id` where competitor_price_history.account_id = ACCOUNT_ID and `competitor_price_history`.`created_at` >= 'ATA 3A ПОСЛЕДНИЕ СУТКИ' order by `competitor_price history`.`created at` desc limit 25

Expected1. Отображаются последние 25 изменений цен конкурентов за последние 24 часа.

result 2. Значения на странице аналогичны выборке базе данных.

Step 8

Input data

Action 1. E

1. Выбрать период "Last month".

2. Выполнить запрос в БД.

select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domain`, `competi tor_price_history`.`clear_new_price` as `new_competitor_price`,ROUND((100 * (CALC_PRICE_WITH_QUANTITY(s phinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.cle ar_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competit ors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference`, `competitors`.`availability` from `competitor_price_history` inner join competit ors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitors`.`id` inner join matchings on `matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_product s`.`id` where competitor_price_history.account_id = ACCOUNT_ID and `competitor_price_history`.`created_at` >= 'ATA 3A ПОСЛЕДНИЙ МЕСЯЦ' order by `competitor_price_history`.`created_at` desc limit 25

Expected. Отображаются последние 25 изменений цен конкурентов за последний 1 месяц.

result 2. Значения на странице аналогичны выборке базе данных.

Step 9

Action

Input

data

1. Отфильтровать страницу по любому домену

2. Выполнить запрос в БД.

select `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domain`, `competi tor_price_history`.`clear_new_price` as `new_competitor_price`,ROUND((100 * (CALC_PRICE_WITH_QUANTITY(s phinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.cle ar_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competit ors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference`, `competitors`.`availability` from `competitor_price_history` inner join competit ors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitor_price_history`.`domain_id` = `domains`.`id` inner join matchings on `matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_product s`.`id` where competitor_price_history.account_id = ACCOUNT_ID and `competitors`.`domain_id` in (DOMAIN_ID and `competitor_price_history`.`created_at` >= 'AATA_3A_ПОСЛЕДНИЙ МЕСЯЦ' order by `competitor_price_history`.`created_at` desc limit 25

Expected. Отображается отфильтрованная выборка за месяц по домену.

result 2. Значения на странице аналогичны выборке базе данных.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Domains

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по домену.

Severity	Priority	Behavior	Туре
☆ Critical	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любого домена и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах и их конкурентах с выбранного домена, кото рые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;
- Expected result
- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.'clear_price` as `own_price`, `domains`.'domain n', `competitor_price_history`.'clear_old_price` as `old_competitor_price_h; `competitor_price_history`.'clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.'created_at`, REPLACE(competitors.availability, " ", " ") as availability_cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_pric e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitors'.`id` inner join domains on `competitors'.`domain_id` = `domains'.`id` inner join matchings on `matchings'.`competitor_id` = `competitor_price_history'.`competitor_id` inner join sphinx_products on `matchings'.`product_id` = `sphin x_products'.`id` where competitor_price_history.account_id =account_idand `competitor_price_history'.`created_at` >= 'Дата/врем Я СУТКИ НазаД' and `competitors'.`domain_id` in (domain_id) and `competitors'.`domain_id` in (domain_id) order by `competitor

Input data

Expected 1. Данные совпадают.

price history`.`created at` desc

Step 3

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах и их конкурентах с выбранного домена, кото рые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit $or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl$ $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinal or a competitor_id` inner join sphinx_products on `matchings`.`products on `m$ x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Aata/Bpem Я МЕСЯЦ Назад' and `competitors`. `domain id` in (domain id) and `competitors`. `domain id` in (domain id) order by `competito r_price_history`.`created_at` desc

result

Expected 1. Данные совпадают.

5 Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах и их конкурентах с выбранного домена, кото рые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах;
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | as `new_price_difference`, `competitor_availa$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc $ount_id = \textbf{account_id} \text{and `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= \texttt{account_id} \text{ and `competitor_availability_history`.} \text{ } \textbf{account_id} \text{ } \textbf{account_id}$ 'дата/время месяц назад' and `competitors`.`domain_id` in (domain_id) and `competitors`.`domain_id` in (domain_id) order by `competitor_availability_history`.`created_at` desc

 $\stackrel{\mathsf{Expected}}{\dots}$ 1. Данные совпадают. result

Step 7

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах и их конкурентах с выбранного домена, кото рые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx products`.'id` as `product id`, `sphinx products`.'title`, `sphinx products`.'clear price` as `own price`, `sphinx products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit y, competitors.clear price) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, `competitor availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor id` = `competitor availability history`.`competitor id` inner join sphinx products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe $titor_id` = `competitors`.`id` \ inner\ join\ domains\ on\ `competitors`.`id` \ where\ competitor_availability_history.acc$ ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'дата/время сутки назад' and `competitors`.`domain_id` in (domain_id) and `competitors`.`domain_id` in (domain_id) order by `competitor_availability_history`.`created_at` desc

result

Expected 1. Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Brands

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по бренду своего продукта.

Severity	Priority	Behavior	Туре
☆ Critical	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted

1

Pre-conditions

- 1. В аккаунте есть продукты разных брендов с матчами.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любого бренда и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного бренда и их конкурентах, котор ые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;
- Expected result
- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl ass, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantity, competitor price history.clear new pric e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `mat$ $x_products`.`id`$ where competitor_price_history.account_id = account_idand`competitor_price_history`.`created_at` >= 'Aata/Bpem' Я СУТКИ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS null AND brand_id =brand_id) order by `competitor_price_history`.`created_at` desc

result

Expected 1. Данные совпадают.

Ctan	
Sten	

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного бренда и их конкурентах, котор ые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit $or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl$ $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Aata/Bpem Я МЕСЯЦ НаЗаД' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS null AND brand id =brand_id) order by `competitor_price_history`.`created_at` desc;

result

Expected 1. Данные совпадают.

5 Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного бренда и их конкурентах, котор ые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | as `new_price_difference`, `competitor_availa$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'дата/время месяц назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS null AND brand_id = brand_id) order by `competitor_availability_history`.`created_at` desc;

result

Expected 1. Данные совпадают.

7 Step

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного бренда и их конкурентах, котор ые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domains`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | a competitor_availabi |$ lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y`inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe $titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc$ ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'Дата/время сутки назад' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS null AND brand id = brand id) order by `competitor availability history`.`created at` desc;

result

Input

data

Expected 1. Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Product types

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по продукт типу своего продукта.

Severity	Priority	Behavior	Type
☆ Critical	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted

1

Pre-conditions

- 1. В аккаунте есть продукты разных продукт типов с матчами.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step

Action 1. В фильтрах активировать чекбокс любого продукт типа и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного продукт типа и их конкурентах, которые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;
- Expected result
- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `mat$ x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created_at` >= 'Aata/Bpem Я СУТКИ НАЗАД' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS NULL AND product _type_id =product_type_id) order by `competitor_price_history`.`created_at` desc;

Expected 1. Данные совпадают. result

Ctan	
Sten	

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного продукт типа и их конкурентах, которые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit $or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl$ $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Aata/Bpem Я МЕСЯЦ НАЗАД' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS NULL AND product _type_id =product_type_id) order by `competitor_price_history`.`created_at` desc;

result

Expected 1. Данные совпадают.

5 Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного продукт типа и их конкурентах, которые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз:
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | as `new_price_difference`, `competitor_availa$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'дата/время месяц назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND product_type_id =product_type_id) order by `competitor_availability_history`.`created_at` desc;

 $\stackrel{\mathsf{Expected}}{\dots}$ 1. Данные совпадают. result

Step 7

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного продукт типа и их конкурентах, которые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx products`.'id` as `product id`, `sphinx products`.'title`, `sphinx products`.'clear price` as `own price`, `sphinx products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit y, competitors.clear price) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, `competitor availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor id` = `competitor availability history`.`competitor id` inner join sphinx products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe $titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc$ ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'дата/время сутки назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND product_type_id = product_type_id) order by `competitor_availability_history`.`created_at` desc;

result

Expected 1. Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Tags

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по тегу, прикрепленному к своему продукту.

Severity	Priority	Behavior	Туре
☆ Critical	↓ Low	Positive	Functional
Milestone	Automation	Status	Is Muted
August 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. К своим продуктам прикреплены разные теги.
- 3. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 4. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любого тега и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным тегом и их конкурентах, котор ые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", ",") as availability_cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.clear_price) / sphinx_products.clear_price) / sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products_products.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_history.clear_price_

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai

Input data

Expected 1. Данные совпадают.

ory`.`created_at` desc;

Step

3

Action 1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным тегом и их конкурентах, котор ые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `mat$ x products'.'id' where competitor price history.account id =account idand 'competitor price history'.'created at' >= 'Дата/врем Я Месяц назад' and `product_id` in (SELECT product_id FROM products_product_labels WHERE product_id IN (SELECT id FROM product s WHERE account_id =account_idAND deleted_at IS NULL) AND product_label_id =product_label_id) order by `competitor_price_hist

result

Expected 1. Данные совпадают.

ory`.`created at` desc;

Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

Action

- 1. На странице в таблице отображаются данные о продуктах с выбранным тегом и их конкурентах, котор ые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | a competitor_availabi |$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'Дата/время месяц назад' and `product_id` in (SELECT product_id FROM products_product_labels WHERE product_id IN (SELECT id FROM products WHERE account_id = $account_id$ = $account_id$ ADD product_label_id = $product_label_id$ order by `compet is account_id = $account_id$ order by ` itor_availability_history`.`created_at` desc;

Expected 1. Данные совпадают. result

Step

Action 1. На детальной странице Competitor Activity виджета выбрать Last 24 hours.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным тегом и их конкурентах, котор ые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

.`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear price` as `new competitor price`, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantit y, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availabi lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a vailability history`.`old availability` as `old availability`, `competitor availability history`.`created at` from `competitor availability history`. y`inner join matchings on `matchings'.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` $matchings`.`product_id` = `sphinx_products`.`id` inner join competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_availability_history`.`co$ titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'Дата/Время сутки назад' and `product id` in (SELECT product id FROM products product labels WHERE product id IN (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL) AND product_label_id =product_label_id) order by `compet

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products`

result

 $\stackrel{\hbox{\it Expected}}{\dots}$. Данные совпадают.

itor_availability_history`.`created_at` desc;

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Price strategy

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по стратегии цены, примененной к своему продукту.

Severity **Priority** Rehavior **Type** Critical Positive **Functional ↓** Low Milestone **Automation Status** Is Muted No August 1.0 (2023) Manual Actual

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. К своим продуктам применены разные стратегии цен.
- 3. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 4. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любой стратегии и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной стратегией и их конкурентах, к оторые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_pric e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphingartan on `matchings'.`product_id` = `sphingartan on `matchi$ x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created_at` >= 'Aata/Bpem' Я СУТКИ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL AND price_st rategy_id =price_strategy_id) order by `competitor_price_history`.`created_at` desc;

Expected 1. Данные совпадают. result

Step

3

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной стратегией и их конкурентах, к оторые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `mat$ x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created at` >= 'ДаТа/Врем Я МӨСЯЦ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL AND price_st $rategy_id = \\price_strategy_id) \ order \ by \ `competitor_price_history`.`created_at` \ desc;$

result

Expected 1. Данные совпадают.

Step 5

Action 1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной стратегией и их конкурентах, к оторые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах;
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step

6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | a competitor_availabi |$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'дата/время месяц назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND price_strategy_id =price_strategy_id) order by `competitor_availability_history`.`created_at` desc;

 $\stackrel{\mathsf{Expected}}{\dots}$ 1. Данные совпадают. result

Step 7

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной стратегией и их конкурентах, к оторые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx products`.'id` as `product id`, `sphinx products`.'title`, `sphinx products`.'clear price` as `own price`, `sphinx products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit y, competitors.clear price) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, `competitor availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor id` = `competitor availability history`.`competitor id` inner join sphinx products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe $titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc$ ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'дата/время сутки назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND price_strategy_id =price_strategy_id) order by `competitor_availability_history`.`created_at` desc;

result

Expected 1. Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Price position

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по прайс позиции своего продукта.

Price position - показатель цены, основанный на разнице конкурента с самой малой ценой и продуктом (Diff). Есть 5 вариантов :

- Lowest price green-price,
- Highest price red-price,
- Neutral price gray-price
- Semi low price light-green-price,
- Semi high price yellow-price.

Severity	Priority	Behavior	Туре
☆ Critical	↓ Low	Positive	Functional
Milestone	Automation	Status	Is Muted
August 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами, цена которых выше/ниже/равна цене своего продукта.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step	1
Action	1. В фильтрах активировать чекбокс любой прайс позиции и закрыть фильтра.
Input data	
	 На странице в таблице отображаются данные о продуктах с выбранной прайс позицией и их конкурент ах, которые изменили цену за последние 24 часа: Title - название своего продукта; Own Price - цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. $clear_price), 2) \ as `old_price_difference` \ from `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitors \ USE \ INDEX \ (PRIMARY) \ on \ `competitor_price_history` \ inner \ join \ competitor_price_history` \ join \ competi$ history`.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` inner join matchings on ` matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin $x_products`.`id`$ where competitor_price_history.account_id = account_idand`competitor_price_history`.`created_at` >= 'Aata/Bpem' Я СУТКИ НАЗАД' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND matchings_data->>'\$."pr ice_color"' = 'price_color') order by `competitor_price_history`.`created_at` desc

Expected ... Данные совпадают. result

Step 3

Action 1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной прайс позицией и их конкурент ах, которые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`. `title`, `sphinx_products`. `clear_price` as `own_price`, `domains`. `domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_pric e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin x_products`.`id` where competitor_price_history.account_id = account_idand `competitor_price_history`.`created_at` >= 'Aata/Bpem Я МӨСЯЦ Ha3aД' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND matchings_data->>'\$."p rice_color" = 'price_color') order by `competitor_price_history`.`created_at` desc

result

Expected 1. Данные совпадают.

5 Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной прайс позицией и их конкурент ах, которые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах:
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

6 Step

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx products`.'id` as `product id`, `sphinx products`.'title`, `sphinx products`.'clear price` as `own price`, `sphinx products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear price` as `new competitor price`, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantit y, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'Дата/время месяц назад' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND matchings _data->>'\$."price_color"' = 'price_color') order by `competitor_availability_history`.`created_at` desc

Expected 1. Данные совпадают. result

7 Step

Action

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранной прайс позицией и их конкурент ах, которые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах:
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

_data->>'\$."price color"' = 'price color') order by `competitor availability history`. `created at` desc

 $1. \ \ select `sphinx_products`.`id` \ as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` \ as `own_price`, `sphinx_products`.`title`, `sphinx_products`.$ $. `availability` as `own_availability`, REPLACE (sphinx_products. availability, "", "_") as own_availability_class, `domain`, `compet the complex of the c$ itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit y, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availabi $lity_history`.`availability` as `availability`, REPLACE (competitor_availability_history.availability, "", "_") as availability_class, `competitor_a availability_history.availability, "", "_") as availability_class, `competitor_a availability_history.availability, """, "_", "_") as availability_history.availability_histo$ vailability_history`.`old_availability` as `old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability_histor y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc $ount_id = \textbf{account_id} \text{and `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= \texttt{account_id} \text{ and `competitor_availability_history`.} \text{ } \textbf{account_id} \text{ } \textbf{account_id}$ 'Дата/время сутки назад' and `product id` in (SELECT id FROM sphinx products WHERE account id =account idAND matchings

Input data

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Gross margin

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по марджину своего продукта. Есть 5 видов gross_margin_color:

- High margin red-margin,
- Semi high margin yellow-margin,
- Regular margin gray-margin,
- Semi low margin light-green-margin,
- Low margin green-margin

Severity	Priority	Behavior	Туре
☆ Critical	↓ Low	Positive	Functional
Milestone	Automation	Status	Is Muted
August 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с разным гросс марджином и с матчами.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любой гросс марджин и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным гросс марджином и их конкуре нтах, которые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

Expected _{1.}	Данные	совпадают.
result	Данные	совпадают

Step 3

1. На детальной странице Competitor Activity виджета выбрать Last month. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным гросс марджином и их конкуре нтах, которые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;
 - Competitor домен конкурента, у которого были изменения цены;

Expected result

- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai $\verb|n',`competitor_price_history'.`clear_old_price' as `old_competitor_price', `competitor_price_history'.`clear_new_price' as `new_competitor_price', `competitor_price_history'.`clear_new_price' as `new_competitor_price', `competitor_price', `co$ or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price history`.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` inner join matchings on ` matchings`.`competitor id` = `competitor price history`.`competitor id` inner join sphinx products on `matchings`.`product id` = `sphin x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Дата/врем Я МЕСЯЦ НАЗАД' and `product id` in (SELECT id FROM sphinx products WHERE account id =account idAND stored properties->>'\$." gross margin color" = 'gross margin color') order by `competitor price history`.`created at` desc

result

Expected . Данные совпадают.

Step 5

Action 1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным гросс марджином и их конкуре нтах, которые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах:
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

select `sphinx_products`.'id` as `product_id`, `sphinx_products`. `title`, `sphinx_products`. `clear_price` as `own_price`, `sphinx_products`. `availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availability_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_availability_history`.`oreated_at` from `competitor_availability_history` inner join matchings on `matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.account_id =account_idand `competitor_availability_history`.`created_at` >= 'Aata/Bpems Mecsil Haaaa' and `product_id` in (Select id From sphinx_products Where account_id =account_idand `competitor_availability_history`.`oreated_at` >= 'Aata/Bpems mergin_color" = 'gross_margin_color" = 'gross_margin_color") order by `competitor_availability_history`.`created_at` desc

Expected. Данные совпадают. result

Step 7

Action 1. На детальной странице Competitor Activity виджета выбрать Last 24 hours.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным гросс марджином и их конкуре нтах, которые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в процентах;
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

.`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availability_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", " ") as availability_class, `competitor_a vailability_history`.`old_availability` as `old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability_history` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount_id =account_idand `competitor_availability_history`.`created_at` >= 'Дата/время сутки назад' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND stored_pro

perties->>'\$."gross_margin_color"' = 'gross_margin_color') order by `competitor_availability_history`.`created_at` desc

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products`

Input data

Expected 1. Данные совпадают. result

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Stock status

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по сток статусу своего продукта.

Severity **Priority Behavior Type** Critical Positive **Functional ↓** Low Milestone **Automation Status** Is Muted No August 1.0 (2023) Manual Actual

1

Pre-conditions

- 1. В аккаунте есть продукты разных сток статусов с матчами.
- 2. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 3. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любого сток статуса своего продукта и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного сток статуса и их конкурентах, к оторые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl $ass, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new$ e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. Input clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price data _history`.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `mat$ $x_products`.`id` \ where \ competitor_price_history.account_id = account_id and `competitor_price_history`.`created_at` >= 'Aata/bpem' account_id accou$ Я СУТКИ НАЗАД' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS NULL AND availabil ity = 'availability (если in stock, то добавить еще availability IS null)) order by `competitor_price_history`. `created_at` d esc:

Expected 1. Данные совпадают.

Step

3

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного сток статуса и их конкурентах, к оторые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

4 Step

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx products`.`id` as `product id`, `sphinx products`.`title`, `sphinx products`.`clear price` as `own price`, `domains`.`domai n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl ass, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantity, competitor price history.clear new pric e) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price history'.'competitor id' = 'competitors'.'id' inner join domains on 'competitors'.'domain id' = 'domains'.'id' inner join matchings on ' matchings`.`competitor id` = `competitor price history`.`competitor id` inner join sphinx products on `matchings`.`product id` = `sphin x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created_at` >= 'Дата/врем Я МЕСЯЦ НаЗаД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL AND availabi lity = 'availability (если in stock, то добавить еще availability IS null)) order by `competitor price history`.`created at`

result

Expected 1. Данные совпадают.

desc:

5 Step

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

Action

- 1. На странице в таблице отображаются данные о продуктах выбранного сток статуса и их конкурентах, к оторые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах:
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or_price`, `competitors`.`availability`, `competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", "_") as availability_cl ass, $ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price_history.clear_n$ e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price history`.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin and inner join sphinx_products on `matchings`.`products on `$ x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Дата/Врем Я МЕСЯЦ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL AND availabi lity = 'availability (если in stock, то добавить еще availability IS null)) order by `competitor_price_history`.`created_at` desc:

result

Expected 1. Данные совпадают.

7 Step

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах выбранного сток статуса и их конкурентах, к оторые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах:
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input

data

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet itors`.`clear price` as `new competitor price`, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantit y, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availabi lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a vailability history`.`old availability` as `old availability`, `competitor availability history`.`created at` from `competitor availability history`. y`inner join matchings on `matchings'.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` $matchings`.`product_id` = `sphinx_products`.`id` inner join competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_availability_history`.`co$ titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'Дата/Время сутки назад' and `product id` in (SELECT id FROM products WHERE account id =account idAND deleted at IS NUL L AND availability = 'availability (если in stock, то добавить еще availability IS null)) order by `competitor_availability_hi story`.`created_at` desc

Expected₁ result

Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Favorite

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по фаворитам своего продукта.

Severity	Priority	Behavior	Туре
☆ Critical	↓ Low	Positive	Functional
Milestone	Automation	Status	Is Muted
August 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. Некоторые продукты добавлены в фавориты.
- 3. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 4. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс фаворита и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах фаворитах, которые изменили цену за посл едние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

2 Step

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_pric e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphingartan on `matchings'.`product_id` = `sphingartan on `matchi$ x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created_at` >= 'Qata/BpeM Я СУТКИ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_iddnd deleted_at IS NULL AND favorite = 'favorite') order by `competitor_price_history`.`created_at` desc;

Expected ... Данные совпадают. result

Step 3

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах фаворитах и их конкурентах, которые изме нили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

= 'favorite') order by `competitor_price_history`.`created_at` desc;

Step 4

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl ass, ROUND((100 * (CALC PRICE WITH QUANTITY(sphinx products.quantity, competitors.quantity, competitor price history.clear new pric e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx products.quantity, competitors.quantity, competitor price history.clear old price) - sphinx products.clear price) / sphinx products. clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphington' inner join sphinx_products on `matchings`.`product_id` = `sphington' inner join sphinx_products on `matchings`.`product_id` = `sphington' inner join sphinx_products on `matchings'.`product_id` = `sphington' inner join sphinx_products on `matchings'.`products on `match$ x products`.`id` where competitor price history.account id =account idand`competitor price history`.`created at` >= 'Aata/BpeM Я МӨСЯЦ НАЗАД' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NULL AND favorite

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai

result

Expected 1. Данные совпадают.

Step 5

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

Action

- 1. На странице в таблице отображаются данные о продуктах фаворитах и их конкурентах, которые изме нили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | a competitor_availabi |$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'дата/время месяц назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND favorite = 'favorite') order by `competitor_availability_history`.`created_at` desc;

Expected

1. Данные совпадают. result

Step 7

Action 1. На детальной странице Competitor Activity виджета выбрать Last 24 hours.

Input data

- 1. На странице в таблице отображаются данные о продуктах фаворитах и их конкурентах, которые изме нили сток статус за последние 24 часа:
 - Title название своего продукта;
- Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

Input data

1. select `sphinx products`.`id` as `product id`, `sphinx products`.`title`, `sphinx products`.`clear price` as `own price`, `sphinx products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", " ") as own_availability_class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit y, competitors.clear_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, `competitor_availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` $matchings`.`product_id` = `sphinx_products`.`id` inner join competitors \ USE \ INDEX \ (PRIMARY) \ on `competitor_availability_history`.$ titor id' = 'competitors'.'id' inner join domains on 'competitors'.'domain id' = 'domains'.'id' where competitor availability history.acc ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'дата/время сутки назад' and `product_id` in (SELECT id FROM products WHERE account_id =account_idAND deleted_at IS NUL L AND favorite = 'favorite') order by `competitor availability history`.`created at` desc

result

Expected 1. Данные совпадают.

Dashboard -> Competitor Activity widget -> Competitor Activity page -> Filters -> Dynamic Tags

Кейс проверяет фильтрацию данных об изменениях цены / сток статуса конкурентов по динамическому тегу, прикрепленному к своему продукту.

Severity **Priority Behavior Type** Critical Positive **Functional ↓** Low Milestone **Automation Status** Is Muted No August 1.0 (2023) Manual Actual

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. К своим продуктам прикреплены разные динамические теги.
- 3. У матчей были изменения цены / сток статуса за последний день и за последний месяц.
- 4. Перейти на детальную страницу Competitor Activity виджета.

Steps to reproduce

Step 1

Action 1. В фильтрах активировать чекбокс любого динамического тега и закрыть фильтра.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным динамическим тегом и их конку рентах, которые изменили цену за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 2

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

c_tag_data->'\$.ids', 'dynamic_tag_id', '\$')) order by `competitor_price_history`.`created_at` desc;

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai n', 'competitor price history'. 'clear old price' as 'old competitor price', 'competitor price history'. 'clear new price' as 'new competit or price`, `competitors`.`availability`, `competitor price history`.`created at`, REPLACE(competitors.availability, " ", " ") as availability cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_pric e) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, ROUND((100 * (CALC PRICE WITH QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products. clear price), 2) as `old price difference` from `competitor price history` inner join competitors USE INDEX (PRIMARY) on `competitor price _history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on ` $matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphingartan on `matchings'.`product_id` = `sphingartan on `matchi$ x_products`.`id` where competitor_price_history.account_id =account_idand `competitor_price_history`.`created_at` >= 'Qata/BpeM Я СУТКИ НАЗАД' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND JSON_CONTAINS(dynami

Expected 1. Данные совпадают. result

Step

3

Action

1. На детальной странице Competitor Activity виджета выбрать Last month.

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным динамическим тегом и их конку рентах, которые изменили цену за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Competitor домен конкурента, у которого были изменения цены;
- New Competitor Price новая цена конкурента + разница между новой ценой конкурента и ценой своег о продукта в процентах + стрелка/показатель вверх/вниз;
- Old Competitor Price старая цена конкурента + разница между старой ценой конкурента и ценой свое го продукта в процентах + стрелка/показатель вверх/вниз;
- Stock status сток статус своего продукта;
- Date/Time дата/время изменения цены конкурента.

Step 4

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

 $\verb|c_tag_data->'\$.ids', 'dynamic_tag_id', '\$')| order by `competitor_price_history`.`created_at` desc; \\$

n`, `competitor_price_history`.`clear_old_price` as `old_competitor_price`, `competitor_price_history`.`clear_new_price` as `new_competitor_price`, `competitors`.`availability,` competitor_price_history`.`created_at`, REPLACE(competitors.availability, " ", ",") as availability_cl ass, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_new_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `new_price_difference`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY (sphinx_products.quantity, competitors.quantity, competitor_price_history.clear_old_price) - sphinx_products.clear_price) / sphinx_products.clear_price), 2) as `old_price_difference` from `competitor_price_history` inner join competitors USE INDEX (PRIMARY) on `competitor_price_history`.`competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join matchings on `matchings`.`competitor_id` = `competitor_price_history`.`competitor_id` inner join sphinx_products on `matchings`.`product_id` = `sphin x products`.`id` where competitor price history.account id =account idand `competitor price history`.`created at` >= 'Дата/врем

Я МӨСЯЦ НАЗАД' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND JSON_CONTAINS(dynami

1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `domains`.`domai

Input

data

Expected result 1. Данные совпадают.

Step 5

1. На детальной странице Competitor Activity виджета активировать **Stock status** радиобаттон.

Input data

Action

- 1. На странице в таблице отображаются данные о продуктах с выбранным динамическим тегом и их конку рентах, которые изменили сток статус за последний месяц:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц ентах;
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 6

Action

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.

1. select `sphinx_products`.'id` as `product_id`, `sphinx_products`.'title`, `sphinx_products`.'clear_price` as `own_price`, `sphinx_products` .`availability` as `own availability`, REPLACE(sphinx products.availability, " ", " ") as own availability class, `domains`.`domain`, `compet itors`.`clear_price` as `new_competitor_price`, ROUND((100 * (CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantit $y, competitors. clear_price) - sphinx_products. clear_price) / sphinx_products. clear_price), 2) \ as `new_price_difference`, `competitor_availabi | a competitor_availabi |$ lity history`.`availability` as `availability`, REPLACE(competitor availability history.availability, " ", " ") as availability class, `competitor a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor_id` = `competitor_availability_history`.`competitor_id` inner join sphinx_products on ` matchings`.`product id` = `sphinx products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor availability history`.`compe titor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where competitor_availability_history.acc ount id =account idand `competitor availability history`.`availability` is not null and `competitor availability history`.`created at` >= 'дата/время месяц назад' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND JSON_CON TAINS(dynamic_tag_data->'\$.ids', 'dynamic_tag_id', '\$')) order by `competitor_availability_history`.`created_at` desc

 $\stackrel{\hbox{\it Expected}}{\dots}$. Данные совпадают. result

Step 7

1. На детальной странице Competitor Activity виджета выбрать Last 24 hours. Action

Input data

- 1. На странице в таблице отображаются данные о продуктах с выбранным динамическим тегом и их конку рентах, которые изменили сток статус за последние 24 часа:
 - Title название своего продукта;
 - Own Price цена своего продукта;

Expected result

- Own stock status сток статус своего продукта;
- Competitor домен конкурента, у которого были изменения сток статуса;
- Competitor price цена конкурента + разница между ценой конкурента и ценой своего продукта в проц
- Comp. stock status новый сток статус конкурента + стрелка/показатель вверх/вниз;
- Date/Time дата/время изменения сток статуса конкурента.

Step 8

Action

Input

data

- 1. Запустить запрос в базе данных, вставив необходимые значения.
- 2. Результат сравнить с отображаемыми данными на странице в таблице.
- 1. select `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`clear_price` as `own_price`, `sphinx_products` .`availability` as `own_availability`, REPLACE(sphinx_products.availability, " ", "_") as own_availability_class, `domains`.`domain`, `compet $itors`.`clear_price` as `new_competitor_price`, ROUND((100*(CALC_PRICE_WITH_QUANTITY(sphinx_products.quantity, competitors.quantity)) and the product of t$ y, competitors.clear price) - sphinx products.clear price) / sphinx products.clear price), 2) as `new price difference`, `competitor availabi lity_history`.`availability` as `availability`, REPLACE(competitor_availability_history.availability, " ", "_") as availability_class, `competitor_a $vailability_history`.`old_availability`, `competitor_availability_history`.`created_at` from `competitor_availability`.$ y` inner join matchings on `matchings`.`competitor id` = `competitor availability history`.`competitor id` inner join sphinx products on ` matchings`.`product_id` = `sphinx_products`.`id` inner join competitors USE INDEX (PRIMARY) on `competitor_availability_history`.`compe titor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` where competitor availability history.acc ount_id =account_idand `competitor_availability_history`.`availability` is not null and `competitor_availability_history`.`created_at` >= 'дата/время сутки назад' and `product_id` in (SELECT id FROM sphinx_products WHERE account_id =account_idAND JSON_CON TAINS(dynamic tag data->'\$.ids', 'dynamic tag id', '\$')) order by `competitor availability history`.`created at` desc

result

Expected 1. Данные совпадают.

Dashboard -> New Prices widget -> Redirecting to the Product info page and back

Redirecting to the Product info page from the Dashboard page and back via clicking on title/gtin

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User.

Account has products with matches.

There are new prices that have been set manually, active price strategy that applied to products.

Product "A" new price set manually, product "B" new price set by "STR1" price strategy.

Dashboard page is opened. "New prices" widget is added.

Steps to reproduce

Step	1
Action	On the "New prices" widget pay attention to the "A" product information. Click on "A" product title.
Input data	
Expected res	Redirecting to "A" Product info page. Sult 2. Compare the "A" product information. 3. Below the "A" product price is the new manual price.
Step	2
Action	Click on the "Back to Dashboard" button.
Input data	
Expected res	sult Dashboard page is opened.
Step	3
Action	On the "New prices" widget click on "A" product GTIN (if the product is deleted the redirect will not happen).
Input data	
Expected res	Redirecting to "A" Product info page. Sult 2. Compare the "A" product information. 3. Below the "A" product price is the new manual price.
Step	4
Action	Click on the "Back to Dashboard" button.
Input data	

Step	5
Action	On the "New prices" widget pay attention to the "B" product information, applied Price strategy. Click on "B" product title.
Input data	
Expected result	 Redirecting to "B" Product info page. Compare the "B" product information. Below the "B" product price is the new price set by strategy. Near the "Price strategy" set "STR1".
Step	6
Action	Click on the "Back to Dashboard" button.
Input data	
Expected result	Dashboard page is opened.
Step	7
Action	On the "New prices" widget click on "B" product GTIN.
Input data	

Redirecting to "B" Product info page.
 Expected result 2. Below the "B" product price is the new price set by strategy.
 Near the "Price strategy" set "STR1".

Expected result Dashboard page is opened.

Dashboard -> New Prices widget -> Redirecting to the Product info page from the New Prices page

Redirecting from New Prices page to the Product info page and back to the Dashboard via clicking on title/gtin

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User.

Account has products with matches.

The strategy "STR1" has been applied to the product "A", and a new manual price has been added to the product "B" in the last 24 hours.

Dashboard page is opened.

"New prices" widget is added.

Steps to reproduce			
Step	1		
Action	On the "New prices" widget click on the "See all new prices".		
Input data	https://app.priceshape.dk/product-list/price-changes		
Expecte result	ed. "New prices" page is opened. 2. Displays a list of products that have a new price set for the last 24 hours.		
Step	1		
Action	Compare with DB.		
Input data	select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphinx_products`.`gtin`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_changes`.`owner` as `owner`, `product_price_changes`.`new_price`, `product_price_changes`.`price_diff` as `diff`, `product_price_changes`.`product_clear_price` as `old_price`, `product_price_changes`.`origin`, `product_price_changes`.`created_at` from `product_price_changes` inner join sphinx_products on `product_price_changes`.`product_id` = `sphinx_products`.`id` left join product_price_strategies on `product_price_changes`.`strategy_id` = `product_price_strategies`.`id` where product_price_changes.account_id = account_ld and `product_price_changes`.`price_diff`!= 0 and `product_price_changes`.`new_price` is not null and `product_price_changes`.`created_at` >= DATE_SUB(NOW(), INTERVAL 24 HOUR) order by `product_price_changes`.`created_at` desc limit 50 offset 0		
Expecte result	ed _{Values} are same.		
Step	2		
Action	On the list pay attention to the "A" product information, applied Price strategy. Click on "A" product title.		
Input data			

1. Redirecting to "A" Product info page. Expected. Compare the "A" product information. 3. Below the "A" product price is the new price set by strategy. result 4. Near the "Price strategy" set "STR1". Step 3 Action Click on the "Back to price changes button". Input data $\label{eq:expected} \textbf{Expected}_{\textbf{The "New prices" page is opened.}}$ result Step 4 Action On the list click on "A" product GTIN. Input data Expected 1. Redirecting to "A" Product info page.
2. Below the "A" product price is the new price set by strategy. result 3. Near the "Price strategy" set "STR1". Step 5 Action Click on the "Back to price changes button". Input data Expected The "New prices" page is opened. result Step 6 On the list pay attention to the "B" product information. Action Click on "B" product GTIN. Input data Expected 1. Redirecting to "B" Product info page. Compare the "B" product information. result 3. Below the "B" product price is the new manual price. 7 Step Click on the "Back to price changes button". Action Input data $\mathsf{Expected}_{\mathsf{The}} \, \mathsf{"New} \, \mathsf{prices"} \, \mathsf{page} \, \mathsf{is} \, \mathsf{opened}.$ result

Step	8
Action	On the list click on "B" product title.
Input data	
Expecte result	1. Redirecting to "B" Product info page. 2. Compare the "B" product information. 3. Below the "B" product price is the new manual price.
Step	9
Action	Click on the "Back to Dashboard" button.
Input data	
Expecte result	ed Dashboard page is opened.

Dashboard -> New Prices widget -> New Prices page -> Search

Find product by using search field on the "New Prices" page.

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User (Admin) to the connected account.

All new prices was set 1 hour ago.

Product "ABCDEF" new price set manually, GTIN 1234567890123.

Product "BBCDEF" new price set by "STR1" price strategy, GTIN 1313131313131.

Dashboard page is opened. "New prices" widget is added.

Steps to reproduce

Step	1
Action	On the "New prices" widget click on the "See all new prices".
Input data	
Expected result	https://app.priceshape.dk/product-list/price-changes. "New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours.
Step	2
Action	Enter the "A" on the search field.
Input data	
Expected result	After entering the list has not changed.
Step	3
Action	Enter the "ABC" on the search field.
Input data	
Expected result	All products whose title starts with "ABC" are displayed.
Step	4
Action	Enter whole "ABCDEF" product title.
Input data	
Expected result	Only the "ABCDEF" product new prices are displayed.

Step	5
Action	Clear the search field.
Input data	
Expected result	Search field is empty.
Step	6
Action	Enter the "13131" on the search field.
Input data	
Expected result	All products whose GTIN starts with "13131" are displayed.
Step	7
Action	Enter whole "13131313131" product GTIN.
Input data	
Expected result	Only the "13131313131" product new prices are displayed.

Dashboard -> New Prices widget -> New Prices page -> Dropdown

Changing product list selection by "Last 24 hours", "Last month" dropdown on "New prices" page.

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User.

There are new prices that have been set manually, active price strategy that applied to products for the last 24 hours and

The "A" product has a new manual price that was set for the last 24 hours.

The "B" product has a new price set by "STR1" price strategy one week ago.

The "C" product has a new price set by "STR2" price strategy 2 months ago.

Dashboard page is opened. "New prices" widget is added.

Steps to reproduce

data

Step	1
Action	On the "New prices" widget click on the "See all new prices".
Input data	https://app.priceshape.dk/product-list/price-changes
Expecte result	 "New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours. On the list pay attention to the "A" product information, "Date/time", User, who set manual price. "B" and "C" products are not displayed.
Step	2
Action	Click on the "Last 24 hours".
Input data	
Expecte result	ed Dropdown is opened. Below the "Last 24 hours" is the "Last month".
Step	1
Action	Check DB.
Input	select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphinx_products`.`gtin`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_changes`.`owner` as `owner`, `product_price_changes`.`new_price`, `product_price_changes`.`price_diff` as `diff`, `product_price_changes`.`product_clear_price` as `old_price`, `product_price_changes`.`origin`, `product_price_changes`.` created_at` from `product_price_changes`.

anges` inner join sphinx_products on `product_price_changes`.` product_id` = `sphinx_products`.` id` left join product_price_strategies on `product_price_changes`.` strategy_id` = `product_price_strategies`.` id` where product_price_changes.account_id = **account_id** and `product_price_changes`.` price_diff` != 0 and `product_price_changes`.` new_price` is not null and `product_price_changes`.` created_at` >= DATE_SUB(NOW(), INTERVAL 24 HOLD) order by `product_price_changes`.` created_at` data !imit_for

HOUR) order by `product_price_changes`.`created_at` desc limit 50

Expecte result	d _{Values} are same.
Step	3
Action	Click on the "Last month".
Input data	
Expecte result	 The list of products has been changed. Displays a list of products that have a new price set for the last moth. On the list pay attention to the "B" product information, "Date/time", "STR1" price strategy that is the source of the new price. "C" product is not diaplayed.
Step	1
Action	Check DB.
Input data	select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphinx_products`.`gtin`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_changes`.`owner` as `owner`, `product_price_changes`.`new_price`, `product_price_changes`.`price_diff` as `diff`, `product_price_changes`.`product_clear_price` as `old_price`, `product_price_changes`.`origin`, `product_price_changes`.`created_at` from `product_price_changes` inner join sphinx_products on `product_price_changes`.`product_id` = `sphinx_products`.`id` left join product_price_strategies on `product_price_changes`.`strategy_id` = `product_price_strategies`.`id` where product_price_changes.account_id = account_id and `product_price_changes`.`price_diff` != 0 and `product_price_changes`.`new_price` is not null and `product_price_changes`.`created_at` >= DATE_SUB(NOW(), INTERVAL 1 MONTH) order by `product_price_changes`.`created_at` desc limit 50
Expecte result	d _{Values} are same.
Step	4
Action	Click on the "Last month".
Input data	
Expecte result	^d Dropdown is opened. Over the "Last month" is the "Last 24 hours".
Step	5
Action	Click on the "Last 24 hours".

1. The list of products has been changed.

Expected. Displays a list of products that have a new price set for the last 24 hours.

result
3. On the list pay attention to the "A" product information, "Date/time", User, who set manual price.
4. "B" and "C" products are not diaplayed.

Dashboard -> New Prices widget -> New Prices page -> Redirecting to the Price strategy edit page

Redirecting from New Prices page to the price strategy editing page via clicking on Source of new price->" Set by price strategy" and on the Price strategy name

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
=	Manual	Actual	No

Pre-conditions

- There are new prices that have been set by price strategies last 24 hours.
 Product "A" new price set by active "STR1" strategy, product "B" new price set by deleted "STR2" price strategy.
- 3. Dashboard page is opened.
- 4. "New prices" widget is added.

Steps to reproduce

Step	1
Action	On the "New prices" widget click on the "See all new prices".
Input data	https://app.priceshape.dk/product-list/price-changes
Expected result	 "New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours.
Step	2
Action	Find the "A" product in the list. Opposite the "A" product click on Source of new price->" Set by price strategy".
Input data	
Expected result	Redirecting to the "STR1" price strategy editing page.
Step	3
Action	Open the Dashboard page. On the "New prices" widget click on the "See all new prices".
Input data	
Expected result	 "New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours.
Step	4
Action	Find the "A" product in the list. Opposite the "A" product click on User/Price strategy-> "STR1" strategy name.
Input data	

Expected result	Redirecting to the "STR1" price strategy editing page.
Step	5
Action	Open the Dashboard page. On the "New prices" widget click on the "See all new prices".
Input data	
Expected result	 "New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours.
Step	6
Action	Find the "B" product in the list. Opposite the "B" product click on Source of new price->" Set by price strategy".
Input data	
Expected result	Nothing happened.
Step	7
Action	Find the "B" product in the list. Opposite the "B" product click on User/Price strategy-> "Strategy is currently deleted".
Input data	
Expected result	Nothing happened.
Step	8
Action	Reproduce test case on the "Last month" tab.
Input data	
Expected result	From "A" product there is redirecting to the "STR1" price strategy editing page. From "B" product redirect not happening to the "STR2" price strategy editing page.

Dashboard -> New Prices widget -> New Prices page -> **Pagination**

Changing the products selection via pages pagination on the "New prices' page.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as a User (Admin) to the connected account.

There are 23 new prices that have been set manually, active price strategy that applied to 25 products for the last month. On Settings page-> User setup-> selected 10 products per page.

Dashboard page is opened. "New prices" widget is added.

Action

Click on the "Right" arrow.

Steps to reproduce			
Step	1		
Action	On the "New prices" widget click on the "See all new prices".		
Input data	https://app.priceshape.dk/product-list/price-changes		
Expecte result	d"New prices" page is opened. Displays a list of products that have a new price set for the last 24 hours.		
Step	2		
Action	On the dropdown choose "Last month".		
Input data			
Expected result	 The list of products has been changed. Displays a list of 10 products that have a new price set for the last moth in order newest first. In the lower left corner, pay attention to the display of the number of products - 1-10 out of 48. In the lower righ t corner, pagination is indicated - 1,2,3,4,5 pages and "Left", "Right" arrows. 		
Step	3		
Action	Click on the "2" page pagination.		
Input data			
Expecte result	1. The list of products has been changed. 2. Displays a list of 10 products that have a new price set for the last moth in order newest first. 3. In the lower left corner, pay attention to the display of the number of products - 10-20 out of 48.		
Step	4		

Input data	
Expecte result	1. The list of products has been changed. 2. Displays a list of 10 products that have a new price set for the last moth in order newest first. 3. In the lower left corner, pay attention to the display of the number of products - 20-30 out of 48.
Step	5
Action	Click on the "5" page pagination.
Input data	
Expecte result	1. The list of products has been changed. 2. Displays a list of 8 products that have a new price set for the last moth in order newest first. 3. In the lower left corner, pay attention to the display of the number of products - 40-48 out of 48
Step	6
Action	Click on the "Left" arrow.

Expected. The list of products has been changed.
2. Displays a list of 10 products that have a new price set for the last moth.
3. In the lower left corner, pay attention to the display of the number of products - 30-40 out of 48.

Dashboard -> New Prices widget -> New Prices page -> Filters, "Clear filters" button and change time period

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. В случае если за один из период нет выборки то в виджете отобразится "No records found" После смены периода отображения все выбранные ранее фильтра не сбрасываются.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

Pre-conditions

- 1. Перейти на New Prices страницу. {Host}/product-list/price-changes
- 2. Период отображения времени установленный на Last 24 hours
- 3. В аккаунте есть:
 - товар №1 (price strategy No Price Strategy, brand clark) с новой ценой неделю назад;
 - товар №2 (price strategy No Price Strategy, brand clark) с новой ценой сегодня;

 - товар №3 (price strategy test, brand tradition) с новой ценой сегодня; товар №4 (price strategy test, brand tradition) с новой ценой 2 месяца назад.

Steps to reproduce

Step	1
Action	Нажать на "Filters" кнопку.
Input data	
Expected result	 Открылась страница фильтров в "Brands" меню. Отображаются неактивные "tradition"(1) и "clark"(1) чекбоксы. "Filter selection" панель не отображается.
Step	2
Action	Активировать "tradition"(1) чекбокс.
Input data	
Expected result	 "tradition"(1) чекбокс стал активным. Появилась "Filter selection" панель. В панели отображается: Brands tradition
Step	3
Action	Закрыть интерфейс фильтров
Input data	
Expected result	 Интерфейс фильтров закрылись. Отображается "Clear filters" кнопка.

Step	4
Action	Выбрать период времени
Input data	Last month
Expected result	1. Кнопка "Clear filters" осталась активной 2. Период времени изменился с Last 24 hours на Last month
Step	5
Action	Нажать на "Clear filters" кнопку.
Input data	
Expected result	1. "Clear filters" кнопка исчезла.
Step	6
Action	Нажать на "Filters" кнопку.
Input data	
Expected result	 Открылась страница фильтров в "Brands" меню. Отображаются неактивные "tradition"(1) и "clark"(1) чекбоксы. "Filter selection" панель не отображается.

Dashboard -> New Prices widget -> New Prices page -> Check data in widget

Проверка значений в "New price" виджете:

- свернутое состояние
- "NEW PRICES" страница, "Last 24 hours"
- "NEW PRICES" страница, "Last month"
- Отфильтрованный виджет "Last month"

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. В настройках выбрано отображать "25 items per page".
- 3. За период в последние 30 дней были изменения цен продуктов посредством мануальных цен и цен от
- 4. На Dashboard странице добавлен "New prices" виджет.
- 5. Открыть Dashboard страницу.

Steps to reproduce

Step	1
Action	Обратить внимание на "New prices" виджет. Сравнить значения в колонках с БД.
Input data	select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphin x_products`.`gtin`, `product_price_strategies`.id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_c hanges`.`owner` as `owner`, `product_price_changes`.`product_price_changes`.`product_price_diff` as `diff`, `product_price_changes`.` product_lear_price` as `old_price`, `product_price_changes`.` origin`, `product_price_changes`.` created_at` from `product_price_changes` in ner join sphinx_products on `product_price_changes`.` product_id` = `sphinx_products`.` id` left join product_price_strategies on `product_price_changes`.` strategy_id` = `product_price_strategies`.` id` where product_price_changes.account_id =account_idand `product_price_changes`.` price_diff`!= 0 and `product_price_changes`.` new_price` is not null order by `product_price_changes`.` created_at` desc limit 5

Expected. Отображаются последние 5 изменений цен продуктов.

result 2. Значения в колонках сходятся со значениями в БД.

Step

Нажать на "See all new prices".

Сравнить значения в колонках с БД.

В поле `product price changes`.`created at` >= '2023-05-03 14:31:14УКазать дату 2023-05-03 - ЧТО соответствует вчера,

а время 14:31:14 -сейчас (сегодня) по UTC.

Input data

Action

 $select `product_price_changes'.`id` as `id`, `sphinx_products'.`id` as `product_id`, `sphinx_products'.`title`, `sphinx_products'.`link`, `sphinx_products'.`title`, `sphinx_products$ x_products .`gtin`, `product_price_strategies`. id` as `strategy_id`, `product_price_strategies`. `name` as `strategy_name, `product_price_changes`.` hanges`.` owner` as `owner`, `product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` origin`, `product_price_changes`.` created_at` from `product_price_changes` in ner join sphinx_products on `product_price_changes`.` product_id` = `sphinx_products'.` id` left join product_price_strategies on `product_price_changes`.` strategy_id` = `product_price_strategies`.` id` where product_price_changes`. account_idand `product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`.` product_price_changes`. product_price_changes`.` product_price_changes`. s`.`price_diff` != 0 and `product_price_changes`.`new_price` is not null and `product_price_changes`.`created_at` >='2023-05-03 14:31:14 'order by `product_price_changes`.`created_at` desc limit 25 offset 0

1. Открылась страница "NEW PRICES".

Expected. Дефолтно (если не было сохраненных фильтров) отображаются изменения цен продуктов за последние 24 часа. result

3. Значения в колонках сходятся со значениями в БД.

Step

3

Выбрать период "Last month".

Action

Сравнить значения в колонках с БД.

В поле `product_price_changes`.`created_at` >= '2023-05-03 14:31:14указать дату 2023-05-03 - ЧТО соответствует 30 дня м назад, а время 14:31:14 -сейчас (сегодня) по UTC.

Input data

select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphinx x_products`.`gtin`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_c hanges`.`owner` as `owner`, `product_price_changes`.`new_price`, `product_price_changes`.`price_diff` as `diff`, `product_price_changes`.` product_clear_price` as `old_price`, `product_price_changes`.` origin`, `product_price_changes`.` created_at` from `product_price_changes` in ner join sphinx_products on `product_price_changes`.` product_id` = `sphinx_products'.`id` left join product_price_strategies on `product_price_changes`.` strategy_id` = `product_price_strategies`.`id` where product_price_changes.account_id=account_idand `product_price_changes.` price_diff` != 0 and `product_price_changes`.` new_price` is not null and `product_price_changes`.` created_at` >= '2023-04-04 14:35:13' order by `product_price_changes`.` created_at` desc limit 25 offset 0

Expected. Виджет отображает изменения цен за последние 30 дней.

result 2. Значения в колонках сходятся со значениями в БД.

Step 4

Отфильтровать виджет по "Source->Strategy".

Action

Сравнить значения в колонках с БД.

В поле `product_price_changes`.`created_at` >= '2023-05-03 14:31:14указать дату 2023-05-03 - что соответствует 30 дня м назад, а время 14:31:14 -сейчас (сегодня) по UTC.

Input data

select `product_price_changes`.`id` as `id`, `sphinx_products`.`id` as `product_id`, `sphinx_products`.`title`, `sphinx_products`.`link`, `sphin x_products`.`gin`, `product_price_strategies`.`id` as `strategy_id`, `product_price_strategies`.`name` as `strategy_name`, `product_price_c hanges`.`owner` as `owner`, `product_price_changes`.`new_price', `product_price_changes`.`price_diff` as `diff`, `product_price_changes`.` product_clear_price` as `old_price`, `product_price_changes`.`product_price_changes`.` created_at` from `product_price_changes` in ner join sphinx_products on `product_price_changes`.` product_id` = `sphinx_products'.` id` left join product_price_strategies on `product_price_changes`.` strategy_id` = `product_price_strategies`.` id` where product_price_changes.account_id = account_id and `product_price_changes`.` price_diff`!= 0 and `product_price_changes`.` new_price` is not null and `origin` in ('new_strategy_price') and `product_price_changes`.` created_at` >= '2023-04-04 14:36:34' order by `product_price_changes`.` created_at` desc limit 25

Expected. Виджет отображает изменения цен стратегиями за последние 30 дней.

result 2. Значения в колонках сходятся со значениями в БД.

Dashboard -> New Prices widget -> New Prices page -> Filters -> Direction

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с атрибутом New price за установленная в такие периоды времени : 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов
 - 3)Меньше 24 часов
- 3. Также у продуктов еть как повышение цены так и понижение. Разница между New_price и Price (цена продукта)
- 4. Выбранный Last 24 hours и открыт интерфий фильтров

Steps to reproduce

Cton

Step	1
Action	Открыть Direction меню и отфильтровать выборку
Input data	1. Up
Expected result	1. Открыто меню Direction 2. Чекбокс Manual стал активный
Step	2
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	SELECT * FROM `product_price_changes` WHERE account_id =account_id and product_clear_price < new_price AND created_at >= NOW () - INTERVAL 24 HOUR;
Expected result	 Интерфейс фильтров закрылся и отображается кнопка "Clear filter" Количестов записей совпало в виджете и в запросе Данные отдельно выбранных продуктов совпадают с данными в базе
Step	3
Action	Сброить фильтра, выбрать "Last month" и перейти на страницу фильтров.
Input data	
Expected result	 Кнопка "Clear filter' не отображается "Last month" опция выбрана. Открытый интерфейс фильтров

Step	4
Action	Открыть Direction меню и отфильтровать выборку
Input data	1. Down
Expected result	1. Открыто меню Direction 2. Чекбокс Strategy стал активный
Step	5
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	SELECT * FROM `product_price_changes` WHERE account_id =account_id and product_clear_price > new_price AND created_at >= NOW () - INTERVAL 30 DAY;
Expected result	 Интерфейс фильтров закрылся и отображается кнопка "Clear filter" Количестов записей совпало в виджете и в запросе Данные отдельно выбранных продуктов совпадают с данными в базе

Dashboard -> New Prices widget -> New Prices page -> Filters -> Mix of filters

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. После смены опции все выбранные ранее фильтра сбрасываются.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No
1	2	3	

Pre-conditions

- 1. Открыта страница {Host}/product-list/price-changes
- 2. Период выбранный Last month
- 3. Открыта страница фильтров
- 4. В аккаунте у продуктов есть изменение новой ценны за месяц

Step	1
Action	Отфильтровать выборку по бренду
Input data	<your_brand></your_brand>
Expecte result	d 1. Чекбокс бренда <your_brand> стал активный</your_brand>
Step	2
Action	Перейти в "Product types" меню и активировать чекбокс
Input data	<your_product_type></your_product_type>
Expecte result	d 1. Чекбокс бренда <your_product_type> стал активный</your_product_type>
Step	3
Action	Сохранить фильтра и сравнить результат с запросом
Input data	SELECT * FROM `product_price_changes` WHERE account_id = <account_id> and product_id IN(SELECT id FROM products WHERE account_id = <account_id> and product_type_id IN (SELECT id FROM product_types WHERE display_delivery = '<your_product_type>') and brand_id IN (SELECT id FROM brands WHERE name = <your_brand>)) and new_price is not null and created_at >= NOW()-INTERVAL 30 DAY and new_price is not null;</your_brand></your_product_type></account_id></account_id>
Expecte result	о 1. Данные в таблици совпадают с данными в запросе

Dashboard -> New Prices widget -> New Prices page -> Filters -> Source

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с атрибутом New price установленные (мануально и стратегией) за установленная в такие периоды времени:
 - 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов
 - 3)Меньше 24 часов
- 3. Выбранный Last 24 hours и открыт интерфий фильтров

Step	1
Action	Открыть Source меню и отфильтровать выборку
Input data	1. Manual
Expecte result	edl. Открыто меню Source 2. Чекбокс Manual стал активный
Step	2
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	select count(`product_price_changes`.`id`) as aggregate from `product_price_changes` inner join sphinx_products on `product_price_changes `.`product_id` = `sphinx_products`.`id` left join product_price_strategies on `product_price_changes`.`strategy_id` = `product_price_strategies`.`id` where product_price_changes.account_id = account_id and `product_price_changes`.`price_diff`!= 0 and `product_price_changes`.`new_price` is not null and `origin` in ('new_user_price') and `product_price_changes`.`created_at` >= 'ДАТА_ВРЕМЯ-СУТКИ_НАЗАД';
Expecte result	ed. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе 3. Данные отдельно выбранных продуктов совпадают с данными в базе
Step	3
Action	Сброить фильтра, выбрать "Last month" и перейти на страницу фильтров.
Input data	

Expected 1. Кнопка "Clear filter' не отображается 2. "Last month" опция выбрана. result 3. Открытый интерфейс фильтров

Step 4

Action Открыть Source меню и отфильтровать выборку

Input data

1. Strategy

Expected. Открыто меню Source

result 2. Чекбокс Strategy стал активный

Step 5

Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу Action

Input data

es`.`id` where product_price_changes.account_id = account_id and `product_price_changes`.`price_diff` != 0 and `product_price_changes `.`new_price` $\underline{is\ not}$ null \underline{and} `origin` \underline{in} ('new_strategy_price') \underline{and} `product_price_changes`.`created_at` >= ' $\underline{\mathsf{LATA}}$ _BPEMS-MECSLL_HA ЗАД';

Expected. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе

result

3. Данные отдельно выбранных продуктов совпадают с данными в базе

Dashboard -> New Prices widget -> New Prices page -> Filters -> Price strategy

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с минимум 3-х Price strategy и No Price Strategy и с атрибутом New_price за установленная в такие периоды времени :
 - 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов

3. Открытый интерфейс фильтров

- 3)Меньше 24 часов
- 3. Выбранный Last 24 hours и открыт интерфий фильтров

меню и отфильтровать выборку
strategy trategy стал активный
ильтров и сравнить полученные данные в выборке с данными по запросу
e_changes` WHERE `account_id` = u >account_id /u > and product_id IN(SELECT id FROM products WHERE d and price_strategy_id IN (SELECT id FROM `product_price_strategies` WHERE account_id = u >accou> r our_PriceStrategy /u >')) AND created_at >= NOW() - INTERVAL 24 HOUR and new_price is not null;
в закрылся и отображается кнопка "Clear filter" совпало в виджете и в запросе ыбранных продуктов совпадают с данными в базе
рать "Last month" и перейти на страницу фильтров.
-

Step 4 Action Открыть Price strategy меню и отфильтровать выборку Input 1. No Price Strategy data Expected. Открыто меню Price strategy result 2. Чекбокс No Price Strategy стал активный 5 Step Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу SELECT * FROM `product_price_changes` WHERE `account_id` = **<u>account_id</u>** and product_id IN(SELECT id FROM products WHERE Input account id =<u>account id</u>and price strategy id IN (SELECT id FROM `product price strategies` WHERE account id =<u>account data nt_id</u> and name ='<u>No Price Strategy</u>' or name ='')) AND created_at >= NOW() - INTERVAL 30 DAY and new_price is not n Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе result 3. Данные отдельно выбранных продуктов совпадают с данными в базе Step 6 Action Открыть интерфейс фильтров и добавить ещё одину Price strategy Input 1. Your PriceStrategy2 data Expected_{1. Открыто меню Price strategy} result 2. Чекбокс Your PriceStrategy2 стал активный

Step 7

Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу

 ${\tt SELECT*FROM`product_price_changes`WHERE`account_id`=<{\tt U>account_id</{\tt IN}(SELECT\ id\ FROM\ products\ WHERE\ account_id)}}$ Input account id =<u>account id</u>and price strategy id IN (SELECT id FROM `product price strategies` WHERE account id =<u>account data $nt_id</u>$ and name =' $<u>No Price Strategy</u>' or name ='<math><u>Your_PriceStrategy2</u>')) AND created_at >= NOW() - NOW() -$ INTERVAL 30 DAY and new_price is not null;

Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей соврадо в сустими. result

3. Данные отдельно выбранных продуктов совпадают с данными в базе

Dashboard -> New Prices widget -> New Prices page -> Filters - > Tags

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No
1	2	2	3

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с минимум 3-х Tags и без тегов (No tag) и с атрибутом New_price за установленная в такие периоды времени:
 - 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов
 - 3)Меньше 24 часов
- 3. Выбранный Last 24 hours и открыт интерфий фильтров

Step	1
Action	Открыть Tags меню и отфильтровать выборку
Input data	1.Yor_Tag
Expecte result	dl. Открыто меню Tags 2. Чекбокс Yor_Tag стал активный
Step	2
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	SELECT * FROM `product_price_changes` WHERE id IN (SELECT id FROM `sphinx_products` WHERE account_id =account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT JSON_ARRAYAGG(id) FROM `product_labels` WHERE account_id =account_id and name = 'Your_Tag'))) and created_at >= NOW() -INTERVAL 24 HOUR;
Expecte result	d. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе 3. Данные отдельно выбранных продуктов совпадают с данными в базе
Step	3
Action	Сброить фильтра, выбрать "Last month" и перейти на страницу фильтров.
Input data	
Expecte result	d. Кнопка "Clear filter' не отображается 2. "Last month" опция выбрана. 3. Открытый интерфейс фильтров

Step Action Открыть Tags меню и отфильтровать выборку Input 1. No tag data Expected. Открыто меню Tags result 2. Чекбокс No tag стал активный 5 Step Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу Input SELECT * FROM `product_price_changes` WHERE id IN (SELECT id FROM `sphinx_products` WHERE account_id =account_id and labels_data is not null and created_at >= NOW() -INTERVAL 30 DAY; data Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе result 3. Данные отдельно выбранных продуктов совпадают с данными в базе Step 6 Открыть интерфейс фильтров и добавить ещё один Тад Action Input 1.Yor_Tag2 data Expected. Открыто меню Tags 2. Чекбокс Yor Tag2 стал активный result 7 Step Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу SELECT * FROM `product_price_changes` WHERE account_id = ```account_id``` and product_id IN (SELECT id FROM `sphinx_products` WHER E account_id =account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data is not null and JSON_CONTAINS(labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id and labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id account_id account_id and labels_data->>'\$."ids"', (SELECT_JSON_ARRAYAGG(id) FROM `productary account_id account_i Input t_labels` WHERE account_id = ```account_id and name ='1'))) and created_at >= NOW() -INTERVAL 30 DAY and new_pri data ce is not null;

result

Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе

3. Данные отдельно выбранных продуктов совпадают с данными в базе

Dashboard -> New Prices widget -> New Prices page -> Filters -> Product types

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с минимум 3-х Product types и No product typeи с атрибутом New_price за установленная в такие периоды времени :
 - 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов
 - 3)Меньше 24 часов
- 3. Выбранный Last 24 hours и открыт интерфий фильтров

Step	1
Action	Открыть Product type меню и отфильтровать выборку
Input data	1.Your_ProductType
Expecte result	dl. Открыто меню Product type 2. Чекбокс Your_ProductType стал активный
Step	2
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	SELECT * FROM `product_price_changes` WHERE `account_id` = <u>account_id</u> and product_id IN(SELECT id FROM products WHERE account_id = <u>account_id=(u)</u> and product_type_id IN (SELECT id FROM product_types WHERE display_name = <u>Your_ProductTy</u> pe ')) AND created_at >= NOW() - INTERVAL 24 HOUR <u>and</u> new_price <u>isnot</u> null;
Expecte result	1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе 3. Данные отдельно выбранных продуктов совпадают с данными в базе
Step	3
Action	Сброить фильтра, выбрать "Last month" и перейти на страницу фильтров.
Input data	
Expecte result	d. Кнопка "Clear filter' не отображается 2. "Last month" опция выбрана. 3. Открытый интерфейс фильтров

Step 4 Action Открыть Product type меню и отфильтровать выборку Input 1.No product type data Expected. Открыто меню Product type result 2. Чекбокс No product type стал активный 5 Step Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу SELECT * FROM `product_price_changes` WHERE `account_id` = < u>account_id </u> and product_id IN(SELECT id FROM products WHERE Input account id = <u>account id </u> and product type id IN (SELECT id FROM product types WHERE display name = <u>No product type data e</u>')) AND created at >= NOW()-INTERVAL 30 DAYand new price isnot null; Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе result 3. Данные отдельно выбранных продуктов совпадают с данными в базе 6 Step Открыть интерфейс фильтров и добавить ещё один Product type Action Input 1.Your ProductType2 data Expected. Открыто меню Product type 2. Чекбокс Your_ProductType2 стал активный result Step 7 Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу Action SELECT * FROM `product_price_changes` WHERE `account_id` = **<u>account_id</u>** and product_id IN(SELECT id FROM products WHERE Input account id = <u>account_id</u> and product type id IN (SELECT id FROM product types WHERE display name = <u>Your_ProductTy data pe2</u>' or display_name = '<u>No product type</u>')) AND created_at >= NOW()-INTERVAL 30 DAYand new_price isnotnull;

Expected 1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе

3. Данные отдельно выбранных продуктов совпадают с данными в базе

result

Dashboard -> New Prices widget -> New Prices page -> Filters -> Brands

Фильтра отображаются в зависимости от выбранной "Last 24 hours" или "Last month" опции. Кейс проверяет верную фильтрацию и отображение верных данных

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
January 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. Открытый развернутый вариант New Prices виджета. Страница {Host}/product-list/price-changes
- 2. В аккаунте есть продукты с минимум 3-х брендов и с атрибутом New_price за установленная в такие периоды времени :
 - 1)Больше 30 дней,
 - 2)Меньше 30 дней но больше 24 часов
 - 3)Меньше 24 часов
- 3. Выбранный Last 24 hours и открыт интерфий фильтров

Step	1
Action	Открыть Brands меню и отфильтровать выборку
Input data	1.Your_brand
	edl. Открыто меню Brands
result	2. Чекбокс стал активный
Step	2
Action	Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу
Input data	SELECT * FROM `product_price_changes` WHERE `account_id` = <u>account_id</u> and product_id IN(SELECT id FROM products WHER E account_id = <u>account_id</u> ')) AND created_at >= NOW() - INTERVAL 24 HOUR <u>and</u> new_price <u>isnot</u> null``;
Expecte result	1. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе совпало. 3. Данные отдельно выбранных продуктов совпадают с данными в базе
Step	3
Action	Сброить фильтра, выбрать "Last month" и перейти на страницу фильтров.
Input data	
Expecte result	ed. Кнопка "Clear filter' не отображается 2. "Last month" опция выбрана. 3. Открытый интерфейс фильтров

Step 4 Открыть Brands меню и отфильтровать выборку Action Input 1.Your_brand2 data Expected. Открыто меню Brands 2. Чекбокс стал активный result 5 Step Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу Action SELECT * FROM `product price changes` WHERE account id <u>account id</u>and product id IN(SELECT id FROM products WHERE a Input ccount_id = <u>account_id</u> and brand_id IN (SELECT id FROM brands WHERE name = <u>Your_brand2</u>')) and new_price is data not null and created_at >= NOW()-INTERVAL 30 DAYand new_price isnotnull``; Expected. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе совпало. result 3. Данные отдельно выбранных продуктов совпадают с данными в базе Step 6 Action Открыть интерфейс фильтров и добавить ещё один Brand Input 1.Your_brand3 data Expected . Открыто меню Brands result 2. Чекбокс стал активный 7 Step Action Закрыть интерфейс фильтров и сравнить полученные данные в выборке с данными по запросу SELECT * FROM `product_price_changes` WHERE account_id =<u>account_id</u>and product_id IN(SELECT id FROM products WHERE a Input ccount id =<u>account_id</u> and brand id IN (SELECT id FROM brands WHERE name =<u>Your_brand2</u> or name =<u>Your_brand2</u> data our_brand3</u>')) and new_price is not null and created_at >= NOW()-INTERVAL 30 DAYand new_price isnotnull``; Expected. Интерфейс фильтров закрылся и отображается кнопка "Clear filter" 2. Количестов записей совпало в виджете и в запросе совпало.

3. Данные отдельно выбранных продуктов совпадают с данными в базе

result

Dashboard->"Stock status" widget-> "All" ratio button

Checking the widget when selected "All" ratio button

SeverityPriorityBehaviorTypeO NormalO MediumPositiveUsabilityMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. Account has products with matches and with filled products_in_stock attribute in the datafeed.
- 2. Dashboard page is opened.
- 3. The "Stock status" widget is added, opened on "Highest" tab.

Step	1
Action	Click on the "All" ratio button.
Input data	
Expecte result	ed. Two sectors with "Highest stock" and "Lowest stock" is displayed. Each sector has columns (max 5). "Highest stock" represents biggest numbers in stock, "Lowest stock" represents smallest numbers in stock.
Step	2
Action	Hover over the columns and compare the displayed number on the column and on the popup.
Input data	
Expecte result	${ m ed}_1$. The numbers on the columns and on the popup are the same.
Step	3
Action	Check values with DB. 1. "Lowest" 2. "Highest"
Input data	SELECT gtin, title, products_in_stock,availability FROM products WHERE account_id =account_idAND products_in_stock is not NULL AND delet ed_at is null AND products_in_stock != 0 ORDER BY products.products_in_stock ASC LIMIT 5 SELECT gtin, title, products in stock,availability FROM products WHERE account_id =account_idAND products in stock is not NULL AND delet

Dashboard->"Stock status" widget-> "Highest" ratio button

Checking the widget displaying when selected "Highest" ratio button

Severity	Priority	Behavior	Type
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches and with filled "products_in_stock" attribute in the datafeed.
- Dashboard page is opened.
 The "Stock status" widget is added.

Step	1
Action	Click on the "Highest" ratio button and check result with database
Input data	SELECT `gtin`, `title`, `products_in_stock` FROM `products` WHERE `account_id` = <u>account_id</u> <u>AND</u> `products_in_stock` <u>is not NULL AND</u> `products_in_stock` != 0 ORDER BY `products_in_stock` DESC LIMIT 10
Expecte result	$^{ m d}$ 1. One sector with "Highest stock" is displayed. There are max 10 columns with the biggest numbers in stock.
Step	2
Action	Hover over the columns and compare the displayed number on the column and on the popup.
Input data	
Expecte result	${ m d}_{ m 1.}$ The numbers on the columns and on the popup are the same.
Step	3
Action	Compare first columns and their numbers on the "Highest" sector when the "All" ratio buton is on with the "Highest" ratio button is on.
Input data	
Expecte result	d1. First columns and numbers on the "Highest" sector/"All" ratio buton match with the columns and numbers on the "Highest" ratio button display.

Dashboard->"Stock status" widget->"Lowest" ratio button

Checking the widget display when selected "Lowest" ratio button

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
2	1	3	1

Pre-conditions

- $1. \ \, Account has products with matches and with filled "products_in_stock" attribute in the datafeed.$
- 2. Dashboard page is opened.3. The "Stock status" widget is added.

Step	1
Action	Click on the "Lowest" ratio button. Compare values with DB.
nput da	ta SELECT `gtin`, `title`, `products_in_stock` FROM `products` WHERE `account_id` = account_id <u>AND</u> `products_in_stock` <u>is not</u> NULL <u>AND</u> `poducts_in_stock` != 0 ORDER BY `products`.`products_in_stock` ASC LIMIT 10
Expecte result	1. One sector with "Lowest stock" is displayed. There are max 10 columns with the smallest numbers in stock.2. Values are same.
Step	2
Action	Hover over the columns and compare the displayed number on the column and on the popup.
Input da	ta
Expecte result	$^{ m J}$ 1. The numbers on the columns and on the popup are the same.
Step	3
Action	Compare first columns and their numbers on the "Lowest" sector when the "All" ratio buton is on with the "Lowest" ratio button is on.
nput da	ta
Expecte esult	d 1. First columns and numbers on the "Lowest" sector/"All" ratio buton match with the columns and numbers on the "Lowest" ratio button display.

Dashboard->"Stock status" widget->"See more" button

Redirecting to Product info page with the selected product by "See more" button

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
2	1	2	3

Pre-conditions

- $1. \ \, Account has products with matches and with filled "products_in_stock" attribute in the datafeed.$
- Dashboard page is opened.
 The "Stock status" widget is added.

Step	1
Action	Click on the "All" ratio button.
Input data	
Expected result	 Two sectors with "Highest stock" and "Lowest stock" is displayed. Each sector has columns (max 5). "Highest stock" represents biggest numbers in stock, "Lowest stock" represents smallest numbers in stock.
Step	2
Action	Hover over one of the columns in the section "Highest stock".
Input data	
Expected result	 Tooltip with "Test" product's title, image, stock status, products in stock and "See more" button is displayed. Pay attention to the data.
Step	3
Action	Click on the "See more" button.
Input data	
Expected result	 Product info page with "Test" product is opened. Pay attention and compare the information on the Product info page with tooltip for "Test" product on the "St ock status" widget. Product's information is the same.
Step	4
Action	Open Dashboard page. On the "Stock status" widget repeat steps №2 and №3 for "All" ratio button "Lowest stock" section, "Highest" ratio button and "Lowest" ratio button.

Input data

1. №2Tooltip with "Test" product's title, image, stock status, products in stock and "See more" button is display ed. Pay attention to the data.

Expected result

- №3Product info page with "Test" product is opened.
 Pay attention and compare the information on the Product info page with tooltip for "Test" product on the "St ock status" widget.
 4. Product's information is the same.

Dashboard ->"Top Domains" widget-> Three radio buttons Switching three ratio-buttons on the "Top Domains" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Usability
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. Account has products with matches from minimum 15 different domains.
- Dashboard page is opened.
 "Top Domains" widget is added.

Step	1
Action	On the "Top Domains" widget click on the second ratio button.
Input data	
Expected result	Widget display was changed from the first page (top 1-5 domains with the biggest number of matches) to the second page (top 6-10).
Step	2
Action	On the "Top Domains" widget click on the third ratio button.
Input data	
Expected result	Widget display was changed from the second page (top $6-10$ domains with the biggest number of matches) to the third page (top $10-15$).
Step	3
Action	On the "Top Domains" widget click on the first ratio button.
Input data	
Expected result	Widget display was changed from the third page (top 10-15) to the first page (top 1-5 domains with the bigge st number of matches).

Dashboard -> "Top Domains" widget -> Filters -> Mix "Tags", "Dynamic tags" parameters + the missing stock status of a competitor

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

1

- 1. В аккаунте есть товары с матчами.
- 2. Все продукты, у которых есть конкурент "test.dk", имеют "dynamic_tag_1" динамический тег и "tag_1" тег.
- 3. Все эти конкуренты в наличии.
- 4. Ha Dashboard странице добавлен "Top domains" виджет.

2

5. Перейти на Dashboard страницу.

Step	1
Action	1. Отфильтровать "Top domains" виджет по "tag_1" тегу и "dynamic_tag_1" динамическому тегу.
Input data	
Expected resu	lt ¹ . Открылся "Choose what you want to see" попап. 2. " tag_1" и "dynamic_tag" отображаются в «Tags» поле поиска.
Step	2
Action	1. Активировать "Out of stock" чекбокс и сохранить изменения.
Input data	
Expected resu	1. Выбранный чекбокс стал активным. It ² . Отображается «Widget updated successfully" is displayed» сообщение. 3. В виджете отображается "Nothing to show at this moment. Please, try again later" сообщение. 4. Данные не отображаются.

Dashboard -> "Top Domains" widget -> Display saved filter title on the widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

2 1

Pre-conditions

- 1. There are products with matches from a minimum of 5 different domains in the account.
- There are products with a minimum of 3 different Brands, Tags, Product types, Dynamic tags, and Labels.
 The Dashboard page is opened.
- 4. The "Top Domains" widget is added.

Steps to reproduce

Step	1
	1. Filter "Top Domains" widget by:
Action	"Test1" and "Test2" brands,
	 "Tag1" tag, "In stock" and "Out of stock" competitor stock statuses.
Input data	
	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
	 The selected 2 Brand's checkboxes are filled with checkmarks. The selected tag checkbox is filled with checkmarks.
Expected resul	4. The selected stock status checkboxes are filled with a checkmark.
	5. The widgets editing mode was closed.6. The "Widget was successfully updated" information message is displayed.
	o. The Waget was successfully aparted information message is displayed.
Step	2
Action	Hover over the inscription under the "Top domains" name.
Input data	
Expected resul	t1. The tooltip shows the names of all selected filters. t2. Brands: "Test1" and "Test2", Tag- "Tag1".
Step	3
	1. Press the F5 button (reload the page).
Action	 Pay attention to the "Top Domains" widget. Hover over the inscription under the "Top domains" name.
Input data	
Expected recul	1. The page is reloaded. t2. "Top domains" widget filtered by Brands: "Test1" and "Test2", Tag- "Tag1".
	2. The colling what the page of all calculated filters

3. The tooltip shows the names of all selected filters.

Step	4
Action	1. Reproduce test case simillary for other combinations of filters (Product types, Labels, Dynamic tags).
Input data	
Expected re	esult ¹ . After reloading of page the "Top domains" widget represents selected filters. 2. The tooltip shows the names of all selected filters.

Dashboard -> "Top Domains" widget-> Check data in widget (User)

Проверка правильности данных в "Top Domains" виджете

Severity	Priority	Behavior	Туре
O Normal	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами с разных доменов (>= 15).
- 2. Ha Dashboard добавлен "Top Domains" виджет.
- 3. Открыть Dashboard страницу.

Steps to reproduce

·	
Step	1
Action	Проверить "Top domains" отображение 15 доменов в пагинации по 5 доменов на вкладку. Проверить порядок отображения доменов (от наибольшего количества матчей к меньшему).
Input data	
Expecte result	d. Отображается 15 доменов в пагинации по 5 доменов на вкладку. 2. Порядок отображения доменов от наибольшего количества матчей к меньшему.
Step	2
Action	Проверить значения "Top Domains" виджета в дефолтном состоянии с БД
Input data	select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitorsid` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join products USE INDEX (PRIMARY) on `matchings`.`product_id` = `products`.`id` where matchings.account_id =account_idand `competitors`.`del eted_at` is null and `products`.`deleted_at` is null group by `matchings`.`domain_id` order by `count` desc limit 15
Expecte result	d. Значения на виджете сходятся с БД.
Step	3
Action	Отфильтровать "Top Domains" виджет по любому бренду и "In stock" статусу. Проверить значения отфильтрованного виджета "Top Domains" с БД.
Input data	select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matchings`.`competitorsid' = `competitors'.`id' inner join domains on `competitors'.`domain_id' = `domains'.`id' inner join products USE INDEX (PRIMARY) on `matchings`.`product_id' = `products'.`id' where matchings.account_id =account_idand `competitors'.`del eted_at` is null and `products'.`deleted_at` is null and (`products'.`brand_id' in (brand_id)) and (`competitors'.`availability' != 'out of stock 'or `competitors'.`availability' is null) group by `matchings'.`domain_id' order by `count' desc limit 15

result

Expected 1. Значения на виджете сходятся с БД.

Step

4

Action

Повторить проверку с другими фильтрами (по бренду, продукт типу, тегу и 2 лейблам, без выбора сток статусов).

Input data

select ANY_VALUE(domain) as name, matchings.domain_id, COUNT(competitor_id) as count, ROUND(AVG((competitors.clear_price / competit ors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as average_deviation from `matchings` inner join competitors USE INDEX (PRIMARY) on `matchings` .`competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` inner join products USE INDEX (PRIMARY) on `matchings` .`product_id` = `products`.`id` where matchings.account_id = account_idand` competitors`.`del doucts use index (PRIMART) on matchings is producted at its null and 'products' is null and 'products'. 'deleted_at' is null and 'products' is null and 'products' is null and 'products' is null and 'products' product_ty pe_id) and (exists (select 'product_labels') product_labels' product_labels' where matchings.product_id = products_product_labels.product_id and product_label_id IN (product_label_id))) and exists (select 'sphinx_products' id' from 'sphinx_products' where matchings.product_id = sphinx_products.id and (custom_labels_data->"\$.data.custom_label_0" = "custom_label1" or custom_labels_data->"\$.data.custom_label_0" = "custom_label_0" = \$.data.custom_label 1" = "custom_label1" or custom_labels_data-> "\$.data.custom_label 2" = "custom_label1" or custom_labels_data-> "\$.data.custom_label_3" = "custom_label1" or custom_labels_data->"\$.data.custom_label_4" = 'custom_label1" or custom_labels_data->"\$. data.custom_label_0" = "custom_label2" or custom_labels_data->"\$.data.custom_label_1" = "custom_label2" or custom_labels_data->"\$.data.custom_label_2" or custom_labels_data->"\$.data.custom_label_2" or custom_labels_data->"\$.data.custom_label_3" = "custom_label2" or custom_labels_data->"\$.data.custom_label_4" = "custom_label2" or custom_labels_data->"\$.data.custom_label_4" = "custom_label2" or custom_labels_data->"\$.data.custom_label_4" = "custom_label2" or custom_labels_data->"\$.data.custom_label_4" or custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_labels_data->"\$.data.custom_l

result

Expected .. Значения на виджете сходятся с БД.

E2E: Dashboard -> "Top Domains" widget -> Filters -> "Brands"

Display "Top Domains" widget filtered by brands

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	3	

Pre-conditions

- 1. There are products with matches from a minimum of 5 different domains in the account.
- 2. There are products from a minimum of 3 different brands in the account.
- 3. "A" domain has matches with products from "Test1" and "Test2" brands.
- 4. The Dashboard page is opened.5. The "Top Domains" widget is added.

steps t	o reproduce
Step	1
Action	 On the "Top Domains" widget click on the red editing "Pen". Select 2 Brands, "Test1" and "Test2", by clicking on checkboxes.
Input data	
Expecte result	 "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. Below the name of the popup, you can see a filter by your own stock status. Under the stock status filter, filters by product attributes are displayed. A dropdown with Brands was displayed. Selected 2 Brand's checkboxes are filled with checkmarks.
Step	2
Action	 Click on the "In stock" and "Out of stock" checkboxes. Click on the "Save" button.
Input data	
Expecte result	 The "COMPETITOR STOCK FILTER" popup was displayed. Selected stock status checkboxes are filled with a checkmark. The widgets editing mode was closed. "Widget was successfully updated" message is displayed.
Step	3
Action	 Pay attention to the "A" domain data, and the number of matches. Opposite the "A" domain, click on the red arrow to the "Right".
Input data	

1. Redirect to the Product list page filtered by "Test1" and "Test2" brands, "A" domain and "In stock", and "Out of

Expected stock" competitor`s stock status.

result 2. Pay attention to the number of matches, only products from "Test1" and "Test2" brands and the "A" domain are displayed.

4 Step

1. Check DB. Action

Input data

1. select `domain` as `name`, `domains`.`id` as `domain id`, count(competitor id) as count, ROUND(AVG((competitors.clear price / compe $titors. quantity) \ / \ (products. clear_price \ / \ products. quantity) \ * \ 100 \), \ 2 \) \ as \ domain_index, \ ROUND(\ AVG(\ (competitors. clear_price \ / \ competitors. quantity) \) \) \)$ s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND $EX \ (matchings_1_product_id) \ on `matchings`.`product_id` = `products`.`id` \ inner join \ competitors \ USE \ INDEX \ (PRIMARY) \ on `matchings`.` \\$ competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where products.account_id = acco unt idand `competitors`.`deleted at` is null and ((products.brand id IN brand id))) and (`competitors`.`availability`!= 'out of stock' o r`competitors`.`availability` is null or`competitors`.`availability` = 'out of stock') and `products`.`deleted_at` is null group by `domains`. 'id' order by 'count' desc limit 15

result

E2E: Dashboard -> "Top Domains" widget -> Filters -> "Product Types"

Display "Top Domains" widget filtered by Product types

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	3	

Pre-conditions

- 1. There are products with matches from a minimum of 5 different domains in the account.
- 2. There are products from a minimum of 3 different product types in the account.
- 3. "A" domain has matches with products from "Test1" and "Test2" product types.
- 4. The Dashboard page is opened.
- 5. The "Top Domains" widget is added.

Steps to r	reproduce
Step 1	
Action 1. 2.	On the "Top Domains" widget click on the red editing "Pen". Select the "Test1" product type by clicking on the checkbox.
Input data	
Expected $\frac{2}{3}$. result $\frac{2}{4}$.	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. Below the name of the popup, you can see a filter by competitor stock status. Under the stock status filter, filters by product attributes are displayed. A dropdown with Product types was displayed. Selected 2 "Test1" Product type checkbox is filled with a checkmark.
Step 2	
Action	Click on the "In stock" and "Out of stock" checkboxes. Click on the "Save" button.
Input data	
Expected2. result 3.	The "COMPETITOR STOCK FILTER" popup was displayed. Selected stock status checkboxes are filled with a checkmark. The widgets editing mode was closed. "Widget was successfully updated" message is displayed.
Step 3	
	Pay attention to the "A" domain data, and the number of matches. Opposite the "A" domain, click on the red arrow to the "Right".
Input data	

1. Redirect to the Product list page filtered by "Test1" product type, "A" domain and "In stock", and "Out of stock" Expected competitor`s stock status.

result 2. Pay attention to the number of matches, only products from "Test1" product type and the "A" domain are displa

yed.

4 Step

Action 1. Check DB.

Input data

 $1. \ \ select `domain` \ as `name`, `domains`.`id` \ as `domain_id`, \ count(competitor_id) \ as \ count, \ ROUND(\ AVG(\ (competitors.clear_price / competitors.clear_price / competitors.clear_price$ titors.quantity) / (products.clear price / products.quantity) * 100), 2) as domain index, ROUND(AVG((competitors.clear price / competitors. s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` where products.account id = acco unt_idand `competitors`.`deleted_at` is null and ((products.product_type_id IN product_type_id))) and `products`.`deleted_at` is null group by `domains`.`id` order by `count` desc limit 15

Expected 1. The data is identical. result

E2E: Dashboard -> "Top Domains" widget-> Filters-> "Tags"

Display "Top Domains" widget filtered by Tags

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	3	

Pre-conditions

- 1. There are products with matches from a minimum of 5 different domains in the account.
- There are products from a minimum of 3 different tags in the account.
 "A" domain has matches with products from "Test1" and "Test2" tags.
- 4. The Dashboard page is opened.5. The "Top Domains" widget is added.

Steps	to reproduce
Step	1
Action	 On the "Top Domains" widget click on the red editing "Pen". Select 2 tags, "Test1" and "Test2", by clicking on the checkboxes.
Input data	
Expecte result	 "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. Below the name of the popup, you can see a filter by your own stock status. Under the stock status filter, filters by product attributes are displayed. A dropdown with tags was displayed. Selected 2 tag's checkboxes are filled with checkmarks.
Step	2
Action	 Click on the "In stock" and "Out of stock" checkboxes. Click on the "Save" button.
Input data	
Expecte result	 The "COMPETITOR STOCK FILTER" popup was displayed. Selected stock status checkboxes are filled with a checkmark. The widgets editing mode was closed. "Widget was successfully updated" message is displayed.
Step	3
Action	 Pay attention to the "A" domain data, and the number of matches. Opposite the "A" domain, click on the red arrow to the "Right".
Input data	

- 1. Redirect to the Product list page filtered by "Test1" and "Test2" tags, "A" domain and "In stock", and "Out of sto Expected ck" competitor`s stock status.

 result 2. Pay attention to the number of matches, only products from "Test1" and "Test2" tags and the "A" domain are di
- result splayed.

Step 4

Action 1. Check DB.

Input data

1. SELECT `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / comp etitors.quantity) / (products.clear price / products.quantity) * 100), 2) as domain index, ROUND(AVG((competitors.clear price / competit ors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation FROM `products` inner join matchings USE I `.`competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` ````WHERE products.account id $= \textbf{aCCOUNt_id} \\ \textbf{and competitors.deleted_at is null and ((exists (select*from products_product_labels where products.id = products_product)} \\ \textbf{and ((exists (select*from products_product_labels where products.id = products_product)} \\ \textbf{and ((exists (select*from products_product_labels where products.id = products_product)} \\ \textbf{and ((exists (select*from products_product_labels where products_product_labels)} \\ \textbf{and ((exists (select*from products_product_labels))} \\ \textbf{and ((exists (select*from products_product))} \\ \textbf{and ((exists (select*from products_products_products_product))} \\ \textbf{and ((exists (select*from products_p$ _labels.product_id and product_label_id IN (product_label_id №1,product_label_id №2)))) and products.deleted_at is null group b y domains.id order by count desc limit 15

result

E2E: Dashboard -> "Top Domains" widget -> Filters -> "Dynamic tags" parameters

Display "Top Domains" widget filtered by Dynamic Tags

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

1 1

Pre-conditions

data

- 1. There are products with matches from a minimum of 5 different domains in the account.
- 2. There are products from a minimum of 1 dynamic tag in the account.
- 3. "A" domain has matches with products from the "Test1" dynamic tag.
- 4. The Dashboard page is opened.
- 5. The "Top Domains" widget is added.

Steps to reproduce		
Step	1	
Action		On the "Top Domains" widget click on the red editing "Pen". Select the "Test1" dynamic tag, by clicking on the checkboxes.
Input data		
Expecte result	ed ² · 3. 4.	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. Below the name of the popup, you can see a filter by your own stock status. Under the stock status filter, filters by product attributes are displayed. A dropdown with dynamic tags was displayed. Selected 1 dynamic tag's checkbox is filled with checkmarks.
Step	2	
Action		Click on the "In stock" and "Out of stock" checkboxes. Click on the "Save" button.
Input data		
Expecte result	ed2. 3.	The "COMPETITOR STOCK FILTER" popup was displayed. Selected stock status checkboxes are filled with a checkmark. The widgets editing mode was closed. "Widget was successfully updated" message is displayed.
Step	3	
Action		Pay attention to the "A" domain data, and the number of matches. Opposite the "A" domain, click on the red arrow to the "Right".
Input		

1. Redirect to the Product list page filtered by the "Test1" dynamic tag, the "A" domain and "In stock", and "Out of

Expected stock" competitor`s stock status.

result 2. Pay attention to the number of matches, only products from the "Test1" dynamic tag and the "A" domain are di splayed.

4 Step

Action 1. Check DB.

Input data

1. select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / compe titors.quantity) / (products.clear price / products.quantity) * 100), 2) as domain index, ROUND(AVG((competitors.clear price / competitors. s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` where products.account id = acco $unt_id and `competitors`.`deleted_at` is null and (exists (select * from `sphinx_products` where `products`.`id` = `sphinx_products`.`id` = `sp$ $and \ (JSON_LENGTH(JSON_NULLABLE(dynamic_tag_data->>'\$.ids')) > 0 \ and \ IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'\$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->>'$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->-'$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->-'$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->-'$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data->-'$.ids', JSON_ARRAYS_INTERSECT(dynamic_tag_data---'$.ids', JSON_ARRAYS_INTER$ _ARRAY(dynamic_tag_id)) = TRUE))) and (`competitors`.`availability`!= 'out of stock' or `competitors`.`availability` is null or `compet $itors`.`availability` = 'out of stock') \ and `products`.`deleted_at` is null group by `domains`.`id` order by `count` desc limit 15 and the stock of the stock$

result

E2E: Dashboard -> "Top Domains" widget -> Filters- > Mix "Tags" and "Dynamic tags" parameters

Display "Top Domains" widget filtered by Tags and Dynamic tags.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	3	

Pre-conditions

- 1. В аккаунте есть товары с матчами.
- 2. Все продукты, у которых есть конкурент "test.dk", имеют "dynamic_tag_1" динамический тег и "tag_1" тег.
- 3. Ha Dashboard странице добавлен "Top domains" виджет.
- 4. Перейти на Dashboard страницу.

Step	1
Action	1. Отфильтровать "Top domains" виджет по "tag_1" тегу и "dynamic_tag_1" динамическому тегу.
Input data	
Expecte result	dl. Открылся "Choose what you want to see" попап. 2. " tag_1" и "dynamic_tag" отображаются в «Tags» поле поиска.
Step	2
Action	1. Активировать "In stock" и "Out of stock" чекбоксы. 2. Сохранить изменения.
Input data	
Expecte result	1. Отображается "COMPETITOR STOCK FILTER" попап. Ф. Выбранные сток статусы стали активными. 3. "COMPETITOR STOCK FILTER" попап закрылся. 4. Отображается "Widget was successfully updated" сообщение.
Step	3
Action	1. Перейти по "arrow" кнопке домена "test.dk".
Input data	
Expecte result	dl. Отображается отфильтрованные товары по " tag_1" тегу, "dynamic_tag_1" динамическому тегу, "test.dk" домену, а также по выбранным "In stock" и "Out of stock" сток статусам конкурента.

Action 1. Проверить значения виджета по БД.

1. select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / compe titors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors. s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` $competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where products.account_id = aCCO in the competitors'.$ unt idand `competitors`.`deleted at` is null and ((exists (select * from `products product labels` where `products`.`id` = `products pro duct_labels`.`product_id` and product_label_id IN (product_label_id))) and exists (select * from `sphinx_products` where `products`.`id $`= `sphinx_products`.`id` and (JSON_LENGTH(JSON_NULLABLE(dynamic_tag_data->>\'$.ids\')) > 0 \ and \ IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->>\'$.ids\')) > 0 \ and \ IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->\')) > 0 \ and \ IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->\') >$ mic_tag_data->>\\\$.ids\\, JSON_ARRAY(dynamic_tag_id)) = TRUE))) and ((`competitors`.`availability`!= \'out of stock\' or `competitors `.`availability` is null) or `competitors`.`availability` = \'out of stock\') and `products`.`deleted_at` is null group by `domains`.`id` order b y `count` desc limit 15

result

Input

data

 $\stackrel{\mathsf{Expected}}{\dots}$ 1. Данные совпадают.

Step 4

Action

1. В соседней вкладке на Product list странице выбрать в фильтрах тот же "dynamic_tag_1" динамически й тег, " tag_1" тег, "test.dk" домен и "In stock", "Out of stock" сток статусы конкурента.

Input data

Expected 1. Результаты перехода по гиперссылке из виджета "Top domains" и на странице "Product list" совпадают.

E2E: Dashboard ->Top Domains widget -> Filters -> "Imported tags"

Display "Top Domains" widget filtered by Imported Tags

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported_tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	

Pre-conditions

- 1. Account has products with matches from minimum 5 different domains.
- 2. Some products with matches from selected domains belongs to "Test1" and "Test2" imported tags.
- 3. Dashboard page is opened.
- 4. "Top Domains" widget is added.

Click on "Save" button.

Action

Step	1
Action	On the "Selected domains" widget click on the "Edit" button.
Input data	
Expecte result	 "EDIT TOP DOMAINS WIDGET" filter's popup was displayed with text "Choose what you want to see and fill out at least one category." Below is the "Name:" input field. "In stock", "Out of stock" checkboxes. "Brand", "Product type", "Tag", "Label" dropdowns. "Reset to default", "Save" buttons.
Step	2
Action	 Filter "Top Domains" widget by imported tags "Test 1", "Test2". Use search field.
Input data	
Expecte result	Dropdown with imported tags was displayed. 2. Search found "Test1" imported tag. 3. Selected 2 imported tags, checkboxes are filled with checkmarks.
Step	3

Input data

Expected. "EDIT TOP DOMAINS WIDGET" filter's popup closed. 2. "Widget was successfully updated" message displayed result

Step 4

Action

- 1. Pay attention to the "A" domain data, number of matches.
- 2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

result

Expected 1. Redirect to Product list page filtered by "Test1" and "Test2" imported tags and "A" domain.

Number of matches are the same, only products from "Test1" and "Test2" imported tags and matches from "A" domain are displayed.

Step 5

Action Compare values on the "Top Domains" widget with the DB.

Input data

select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((com petitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_ind ex, ROUND(AVG((competitors.clear_price / competitors.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average deviation from 'products' inner join matchings USE INDEX (matchings 1 product id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.`co mpetitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where pro ducts.account_id = account_id and `competitors`.`del eted_at` is null and ((exists (select * from `product_imported_tags` where `products`.`id` = `product_imported tags`.`product id` and imported tag id IN (imported tag ids< /span>)))) and `products`.`deleted at` is null group by `domains`.`id` order by `count` desc limit 15;

Expected Values are the same. result

E2E: Dashboard -> "Top Domains" widget -> Filters -> "Labels"

Display "Top Domains" widget filtered by Custom Labels.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	5	

Pre-conditions

- 1. There are products with matches from a minimum of 5 different domains in the account.
- 2. There are products from a minimum of 3 different custom labels in the account.
- 3. "A" domain has matches with products from "Test1" and "Test2" custom labels.
- 4. The Dashboard page is opened.
- 5. The "Top Domains" widget is added.

Steps to reproduce		
Step	1	
Action	1. Filter the "Top Domains" widget with "Test1" and "Test2" custom labels.	
Input data		
Expecte result	 "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. Below the name of the popup, you can see a filter by your own stock status. Under the stock status filter, filters by product attributes are displayed. A dropdown with Labels was displayed. Selected 2 "Test1" custom labels checkboxes are filled with a checkmark. 	
Step	2	
Action	 Click on the "In stock" and "Out of stock" checkboxes. Click on the "Save" button. 	
Input data		
Expecte result	 The "COMPETITOR STOCK FILTER" popup was displayed. Selected stock status checkboxes are filled with a checkmark. The widgets editing mode was closed. "Widget was successfully updated" message is displayed. 	
Step	3	
Action	 Pay attention to the "A" domain data, and the number of matches. Opposite the "A" domain, click on the red arrow to the "Right". 	
Input data		

1. Redirect to the Product list page filtered by "Test1", and "Test2" custom labels, "A" domain and "In stock", and Expected "Out of stock" competitor`s stock status.

esult 2. Pay attention to the number of matches, only products from "Test1", and "Test2" custom labels and the "A" do main are displayed.

Step

Action 1. Check DB.

4

Input data

1. select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / compe titors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors. s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` competitor_id' = 'competitors'.'id' inner join domains on 'competitors'.'domain_id' = 'domains'.'id' where products.account_id = acco unt_idand `competitors`.`deleted_at` is null and (custom_labels_data LIKE "%custom_label_name №1%" or custom_labels_data LIK E "%custom_label_name №2%") and products.deleted_at is null group by domains.id order by count desc limit 15

result

E2E: Dashboard -> "Top Domains" widget -> "Competitor Stock status"

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
		-	
Milestone	Automation	Status	Is Muted

1 1

Pre-conditions

- 1. There are products with matches from different domains in the account.
- 2. The "A" domain has products with "In stock" and the "B" domain has products with "Out of stock" statuses.
- 3. The Dashboard page is opened.
- 4. The "Top Domains" widget is added.

Steps to reproduce

Step 1

1. Filter the "Top Domains" widget by the "In stock" parameter.

Action

- 2. Don't choose Brands, Labels, Product types, Tags, or Dynamic tags.
- 3. Save changes.

Input data

- 1. "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
- 2. Below the name of the popup, you can see a filter by competitor stock status.

Expecteds. Under the stock status filter, filters by product attributes are displayed.

result

- 4. The selected "In stock" checkbox is filled with a checkmark.
- 5. The widgets editing mode was closed.
- 6. "Widget was successfully updated" message is displayed.

Step 1

1. Check DB. Action

Input data

1. select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / compe titors.quantity) / (products.clear_price / products.quantity) * 100), 2) as domain_index, ROUND(AVG((competitors.clear_price / competitors. s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` competitor id` = `competitors`.`id` inner join domains on `competitors`.`domain id` = `domains`.`id` where products.account id =acco unt_idand `competitors`.`deleted_at` is null and (`competitors`.`availability` != \'out of stock\' or `competitors`.`availability` is null) and `products`.`deleted_at` is null group by `domains`.`id` order by `count` desc limit 15

result

Expected 1. The data is identical.

Step 2

1. Pay attention to the "A" domain data, and the number of matches. Action

2. Opposite the "A" domain, click on the red arrow to the "Right".

Input data

Expected1. Redirect to the Product list page filtered by "A" domain, all matches are In stock. 2. Pay attention to the number of matches, only matches In stock from the "A" domain are displayed. Step 3 1. Reset to default "Top Domains" widget. Action Input data result 4 Step 1. Filter the "Top Domains" widget by the "Out of stock" parameter. 2. Don't choose Brands, Labels, Product types, Tags, or Dynamic tags. Action 3. Save changes. Input data 1. "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed. 2. Below the name of the popup, you can see a filter by competitor stock status. Expected. Under the stock status filter, filters by product attributes are displayed. 4. The selected "Out of stock" checkbox is filled with a checkmark. result 5. The widgets editing mode was closed. 6. "Widget was successfully updated" message is displayed. Step 1 Action 1. Check DB. 1. select `domain` as `name`, `domains`.`id` as `domain_id`, count(competitor_id) as count, ROUND(AVG((competitors.clear_price / compe titors.quantity) / (products.clear price / products.quantity) * 100), 2) as domain index, ROUND(AVG((competitors.clear price / competitor s.quantity) / (products.clear_price / products.quantity) * 100) - 100, 2) as average_deviation from `products` inner join matchings USE IND Input EX (matchings_1_product_id) on `matchings`.`product_id` = `products`.`id` inner join competitors USE INDEX (PRIMARY) on `matchings`.` data $competitor_id` = `competitors`.`id` inner join domains on `competitors`.`domain_id` = `domains`.`id` where products.account_id = account_id = account_id` = account_id`$ by `domains`.`id` order by `count` desc limit 15 result Step 5 1. Pay attention to the "B" domain data, and the number of matches. Action 2. Opposite the "B" domain, click on the red arrow to the "Right". Input data Expected. Redirect to the Product list page filtered by the "B" domain, all matches are out of stock.

2. Pay attention to the number of matches, only matches out of stock from the "B" domain are displayed.

result

E2E: Dashboard -> "Top Domains" widget-> "Reset to default" button

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

2 1

Pre-conditions

- 1. There are products with matches from a minimum of 5 domains, products have a few of each Brands, Product types, Tags, Dynamic tags, and Labels.
- 2. The "Top domains" widget is added and filtered by custom settings (Brands, Product types, Tags, Labels, and stock status were chosen).

Steps to reproduce

```
Step
                      1
Action
                      1. Check result on database.
Input data
                     1. SELECT * FROM `widgets` WHERE `account_id` =account_idand user_id =user_idAND id = <widget_id>
                      1. In the "fields" params is displayed JSON with filters:
                         {
                         "origin": null,
                         "brandsIds": [
                         "labelsIds": [
                         0
                         ],
                         "dynamicTags": [],
                         "customLabels": [
Expected result
                         0,
                         "Seee"
                         ],
                         "productTypesIds": [
                         9822,
                         0,
                         9872
                         "competitorsStockStatus": [
                         "in stock"
                         ]
                         }
Step
                      2
Action
                      1. Open the Dashboard page.
Input data
```

Step 3

Expected result

1. The Dashboard page is opened.

2. The "Top domains" widget is displayed.

Action

1. Open filters in the widget and reset to default the "Top domains" widget.

Input data

- 1. "Choose what you want to see" pop-up is displayed.
- Expected result 2. "Widget was successfully updated" information message is displayed.
 - 3. "All products" is displayed above the list of domains.



Step

4

}

Action

1. Check result on database.

Input data

1. SELECT * FROM `widgets` WHERE `account_id` =account_idand user_id =user_idAND id = <widget_id>

1. In the "fields" params is displayed JSON with filters:

"origin": null, "subTitle": null, "brandsIds": [], "domainIds": [], "labelsIds": [], "customLabels": [], "productTypesIds": [], "competitorsStockStatus": []

Expected result

Dashboard -> Inventory Value widget -> Radio buttons

Changing the "Inventory value" widget display clicking on "Total"/ "Top 5 products"/ "Top 5 brands" radio buttons.

Priority Behavior Severity **Type** O Normal O Medium Not set Regression Milestone Automation **Status** Is Muted Manual Actual No

Pre-conditions

There are more than 5 products with matches in the account. Products have been filled by "Cost_price" and "products_in_stock" attributes. There are products with a minimum of 5 brands.

The Dashboard page is opened.

The "Inventory value " widget is added.

Step	1
Action	On the "Inventory value" widget click on the "Top 5 products" radio button.
Input data	
Expected result	 The "Inventory value" widget display was changed. On the "Top 5 products" tab display the inventory value number for top 5 products with the highest inventory value for the product to the lowest.
Step	2
Action	On the "Inventory value" widget click on the "Top 5 brands" radio button.
Input data	
Expected result	 The "Inventory value" widget display was changed. On the "Top 5 brands" tab display the inventory value number for the top 5 brands with the highest sum inventory value number for the brand to the lowest.
Step	3
Action	On the "Inventory value" widget click on the "Total" radio button.
Input data	
Expected result	 The "Inventory value" widget display was changed. On the "Total" tab display the total inventory value number for all products.

Dashboard -> Inventory Value widget -> Redirecting to the Product info page and back

Redirecting to the Product info page from "Top 5 products" tab on the "Inventory value" widget.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are more then 5 products with matches in the account.
- $2. \ \ Products\ have\ filled\ "cost_price"\ and\ "products_in_stock"\ attribute.$
- 3. Account has products with minimum 5 Brands.
- 4. Dashboard page is opened.
- 5. "Inventory value" widget is added.

Step	1
Action	On the "Inventory value" widget click on the "Top 5 products" radio button.
Input data	
Expected result	 The "Inventory value" widget display was changed. On the "Top 5 products" tab display the top 5 products with the highest inventory value number to the lo west.
Step	2
Action	Pay attention to the "A" product inventory value number. Opposite the "A" product click on the red "Right" arrow. Pay attention to the "A" product inventory value number on the Product info page. Compare "Cost price" * "In stock" = "Inventory value" attributes.
Input data	
Expected result	 Redirecting to the "A" Product info page. Inventory value number on the Product info page is the same as on the "Inventory value " widget for "A" product.
Step	3
Action	Click on the "Back to dashboard" button.
Input data	
Expected result	 Dashboard page is opened. "Inventory value" widget opened on the "Total" tab.

Dashboard -> Inventory Value widget -> Redirecting to the Product list page

Redirecting to the Product list page from "Top 5 brands" tab on the "Profit" widget.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

There are more then 5 products with matches in the account.

- 1. Products have filled "cost_price" and "products_in_stock" attributes.
- 2. Account has products with minimum 5 Brands.
- 3. Dashboard page is opened.
- 4. "Inventory value " widget is added.

Step	1
Action	On the "Inventory value" widget click on the "Top 5 brands" radio button.
Input data	
Expected res	ult 1. The "Inventory value" widget display was changed. 2. On the "Top 5 brands" tab display top 5 brands with the highest inventory value numbers to the lowest.
Step	2
Action	Opposite the "A" brand click on the red "Right" arrow.
Input data	
Expected res	ultRedirecting to the Product list page filtered by "A" brand.
Step	3
Action	Click on the "Filters" button.
Input data	
Expected result 1. Filters interface is opened. 2. On the "Filter selection" "A" brand is choosed.	

E2E: Dashboard -> Inventory Value widget -> Filters -> Brands

"Inventory Value" widget displaying for selected Brand

SeverityPriorityBehaviorTypeO NormalO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
In the datafeed products have "cost_price" and "products_in_stock" atribute.
Account has products from minimum 3 different brands.
Dashboard page is opened.
"Inventory Value" widget added.

Step	1
Action	On the "Inventory Value" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Brands" search field.
Input data	
Expected result	Dropdown with Brands was displayed.
Step	3
Action	Select 2 Brands, "Test1" and "Test2", by clicking on checkboxes.
Input data	
Expected result	Selected 2 Brand's checkboxes are filled with checkmarks.
Step	4
Action	Click on the "Save" button.
Input data	

Expected result	 Widgets editing mode was closed. Information message with "Widget was successfully updated" was sho wn. Wiget is opened on "Total" tab with the total number for "Test1" and "Test2" brands.
Step	5
Action	Click on the "Top 5 brands" ratio button.
Input data	
Expected result	 The "Inventory Value" widget display was changed. On the "Top 5 brands" list there are only "Test1" and "Test2" Brands are represented. The common Inventory Value number "Test1" + "Test2"= total number on the "Total" tab.

E2E: Dashboard -> Inventory Value widget -> Filters -> Product Types

Inventory value widget displaying for selected Product type.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
In the datafeed products have "cost _price" and "products_in_stock" atribute.
Account has products from minimum 2 different product types.
Dashboard page is opened.
"Inventory Value" widget added.

Step	1
Action	On the "Inventory Value" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Product types" search field.
Input data	
Expected result	Dropdown with Product types was displayed.
Step	3
Action	Select "Test1" Product type by clicking on checkbox.
Input data	
Expected result	Selected "Test1" Product type checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	
Expected result	Widgets editing mode was closed.

Step	5
Action	Click on the "Top 5 products" ratio button.
Input data	
Expected result	The "Inventory Value" widget display was changed. On the "Top 5 products" list there are only products with "Test1" Product type.
Step	6
Action	Opposite "A" product click on the red "Arrow".
Input data	
Expected result	Redirecting to the "A" Product info page. Pay attention to the "A" product Product type.

E2E: Dashboard -> Inventory Value widget -> Filters -> Tags

"Inventory Value" widget displaying for selected Tag

SeverityPriorityBehaviorTypeO NormalO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
In the datafeed products have "cost _price" and "products_in_stock" atribute.
Account has products with applied "Tag1", other products have "No tag" filter.
Dashboard page is opened.
"Inventory Value" widget added.

Step	1
Action	On the "Inventory Value" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expected result	Dropdown with Tags was displayed.
Step	3
Action	Select "Test1" Tag by clicking on checkbox.
Input data	
Expected result	Selected Tag's checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	
Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. "Inventory Value" widget is opened on the "Total" tab

Step	5
Action	Click on the "Top 5 products" radio button.
Input data	
Expected result	The "Inventory Value" widget display was changed. On the "Top 5 products" list there are only products with "Test1" Tag.
Step	6
Action	Opposite "A" product click on the red "Arrow".
Input data	
Expected result	Redirecting to the "A" Product info page. Pay attention to the "A" applied tag ("Test1").

E2E: Dashboard -> Inventory Value widget -> Filters -> Dynamic Tags

"Inventory Value" widget displaying for selected Tag

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
In the datafeed products have "cost _price" and "products_in_stock" atribute.
Account has products with applied "DT1", other products have "No Dynamic tags" filter.
Dashboard page is opened.
"Inventory Value" widget added.

Step	1
Action	On the "Inventory Value" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expected result	Dropdown with Tags was displayed.
Step	3
Action	Select "DT1" dynamic tag by clicking on checkbox.
Input data	
Expected result	Selected Tag's checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	

Step	5
Action	Click on the "Top 5 products" radio button.
Input data	
Expected result	 The "Inventory Value" widget display was changed. On the "Top 5 products" list there are only products with "DT1" dynamic tag.
Step	6
Action	Opposite "A" product click on the red "Arrow".
Input data	
Expected result	Redirecting to the "A" Product info page. Pay attention to the "A" applied "DT1" dynamic tag.

Widgets editing mode was closed.
 Expected result 2. Information message with "Widget was successfully updated" was shown.
 "Inventory Value" widget is opened on the "Total" tab.

E2E: Dashboard -> Inventory value widget -> Filters -> "Imported tags"

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported_tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	3

Pre-conditions

- 1. Log in as user/admin/RA to the connected account.
- 2. There are products with matches in the account.
- 3. In the datafeed products have "cost price" and "products in stock" atribute.
- 4. Account has products with applied "ImportedTag1", other products have "No tag" filter.
- 5. Dashboard page is opened.
- 6. "Inventory Value" widget added.

Step	
Action	On the "Inventory Value" widget click on the red editing "Pen".
Input data	
Expecte result	ed _" EDIT INVENTORY VALUE WIDGET" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expecte result	ed Dropdown with Tags was displayed.
Step	3
Action	Click on "Imported Tags" field.
Input data	

Expected Dropdown with Imported Tags are opened. result			
Step	4		
Action	Select "ImportedTag1" imported tag by clicking on checkbox.		
Input data			
Expecte result	d Selected Imported tag checkbox is filled with the checkmark.		
Step	5		
Action	Click on the "Save" button.		
Input data			
Expecte result	1. Widgets editing mode was closed. 2. Information message with "Widget was successfully updated" was shown. 3. "Inventory Value" widget is opened on the "Total" tab		
Step	6		
Action	Run query in DB:		
Input data	SELECT sum(cost_price*products_in_stock) as INVENTORY_VALUE FROM `products` inner join product_imported _tags on products.id = product_imported_tags.product_id inner join imported_tags on product_imported_tags.im ported_tag_id = imported_tags.id where account_id = account_id and imported_tags.name = ' imported_tags_name ' ORDER BY INVENTORY_VALUE DESC;		
Expecte result	d Value in the same with "Total" tab.		
Step	7		
Action	Change tab on "Top 5 products" and run query in DB:		
Input data	SELECT title, cost_price*products_in_stock as INVENTORY_VALUE FROM `products` inner join product_imported_tags on products.id = product_imported_tags.product_id inner join imported_tags on product_imported_tags.im ported_tag_id = imported_tags.id where account_id = account_id and imported_tags.name = ' imported_tags_name ' ORDER BY INVENTORY_VALUE DESC;		
Expecte result	^d Value in the same with "Top 5 products" tab.		
Step	8		
Action	Change tab on "Top 5 brands" and run query in DB:		

Input data SELECT brands.name as BRAND, sum(cost_price*products_in_stock) as INVENTORY_VALUE FROM brands inner j oin `products` on brands.id = products.brand_id inner join product_imported_tags on products.id = product_imp orted_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id w here account_id = account_id and imported_tags.nam e = 'imported_tags_name' GROUP BY brands.name OR DER BY INVENTORY_VALUE DESC;

Expected_{Value} in the same with "Top 5 brands" tab. result

E2E: Dashboard -> Inventory Value widget -> Filters -> Labels

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
Products have filled "cost_price" and "products_in_stock" attribute.
Account has products with minimum 2 different Product labels.
"A" brand has 2 products.
Dashboard page is opened.
"Inventory value " widget is added.

Step	1
Action	On the "Inventory value" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Labels" search field.
Input data	
Expected result	Dropdown with Labels was displayed.
Step	3
Action	Select "Test1" Label by clicking on checkbox.
Input data	
Expected result	Selected "Test1" Label checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	

Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.
Step	5
Action	Click on the "Top 5 brands" radio button.
Input data	
Expected result	The "Inventory value" widget display was changed. On the "Top 5 brands" list there are only products with "Test1" Label.
Step	6
Action	Pay attention to the "A" brand inventory value data. Opposite "A" brand click on the red "Arrow".
Input data	
Expected result	Redirecting to the Product list with products that belong to the "Test1" Label and "A" brand.
Step	7
Action	Click on 2 product's product card. Pay attention to the inventory value numbers. Sum up the inventory value numbers for all (2) products from the "A" brand and "Test1" Label.
Input data	
Expected result	Redirecting to the selected product's Product info page. Pay attention to the Labels attribute. The products h as "Test1" Label. The sum inventory value number is the same as on the "Inventory value" widget.

E2E: Dashboard -> Inventory Value widget -> Filters -> "Reset to default" button

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. All products have filled "cost price" and "products_in_stock" attribute.
- 3. Dashboard page is opened.
- 4. "Inventory value" widget is added and filtered by custom parameters (by Brands, Tags, Dynamic tags, Product types, Labels).

```
Steps to reproduce
Step
Action
                   Run a query in the database
Input data
                   SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
                   In the fields params displayed JSON which displayed your filters:
                   The result in the database is the same as on the page
                   {
"origin": null,
                   "brandsIds": [
                   brand id 1
                   "labelsIds": [
                   label id 1
Expected result ], "dynamicTags": [
                   dynamic_tag_id_1
                   "customLabels": [
                   "name"
                   "productTypesIds": [
                   product_type_id_1
                   2
Step
Action
                   Click on the red pen "Edit" button in the upper left corner on the "Inventory value" widget.
Input data
Expected result "Choose what you want to see" pop-up is displayed.
Step
                   3
Action
                   Click on "Reset to default" button.
Input data
```

Expected result "Widget was successfully updated" information message was displayed.

```
Action Run a query in the database

Input data SELECT * FROM `widgets` WHERE `account_id` =account_id and user_id = user_id AND id = id_your_widget

In the fields params displayed JSON which displayed your filters:
The result in the database is the same as on the page

{
    "origin": null,
    "brandslds": [],
    "dynamicTags": [],
    "customLabels": [],
    "productTypesIds": []
}
```

Dashboard -> Profit widget -> "Last N days" dropdown

Displaying profit for the selected period of time for "Total"/ "Top 5 products"/ "Top 5 brands"

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin to the connected account.

There are more then 5 products with matches in the account.

Products have filled "Cost price" attribute.

Account has products with minimum 5 Brands.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

On the Settings page Google Analytics settings-> Data interval is 30 days.

Dashboard page is opened.

"Profit " widget is added.

data

Steps t	o reproduce
Step	1
Action	On the "Profit" widget click on the black arrow "Down" next to "Last 30 days". Click on the "Last day".
Input data	
Expecte result	 Dropdown with different periods is opened. The "Profit" widget display was changed. On the "Total"/ "Top 5 products"/ "Top 5 brands" tabs display the number of profit for the last day relative to all products / top 5 products with the highest profit to the lowest / top 5 brands with a total profit by product from the highest to the lowest.
Step	2
Action	On the "Profit" widget click on the black arrow "Down" next to "Last day". Click on the "Last 7 days".
Input data	
Expecte result	 Dropdown with different periods is opened. The "Profit" widget display was changed. On the "Total"/ "Top 5 products"/ "Top 5 brands" tabs display the number of profit for the last 7 days relative to all products / top 5 products with the highest profit to the lowest / top 5 brands with a total profit by product from the highest to the lowest.
Step	3
Action	On the "Profit" widget click on the black arrow "Down" next to "Last 7 days". Click on the "Last 14 days".
Input	

Dropdown with different periods is opened.

result

Expected The "Profit" widget display was changed.
On the "Total"/ "Top 5 products"/ "Top 5 brands" tabs display the number of profit for the last 14 days relative to a Il products / top 5 products with the highest profit to the lowest / top 5 brands with a total profit by product from t he highest to the lowest.

4 Step

Action

On the "Profit" widget click on the black arrow "Down" next to "Last 14 days". Click on the "Last 6 months".

Input data

Dropdown with different periods is opened.

Expected The "Profit" widget display was changed.

On the "Total"/ "Top 5 products"/ "Top 5 brands" tabs display the number of profits for the last 6 months relative t o all products / top 5 products with the highest profit to the lowest / top 5 brands with a total profit by product fro m the highest to the lowest.

Dashboard -> Profit widget -> Radio buttons

Changing the "Profit" widget display clicking on "Total"/ "Top 5 products"/ "Top 5 brands" radio buttons

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin/RA to the connected account.

There are more then 5 products with matches in the account.

Products have filled "Cost price" attribute.

Account has products with minimum 5 Brands.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

On the Settings page Google Analytics settings-> Data interval is 30 days.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	On the "Profit" widget click on the "Top 5 products" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 products" tab display the profit number for the last 30 days for the top 5 products with the high est profit to the lowest.
Step	2
Action	On the "Profit" widget click on the "Top 5 brands" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 brands" tab display the profit number for the last 30 days for the top 5brands with the highest ρ rofit to the lowest.
Step	3
Action	On the "Profit" widget click on the "Total" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Total" tab display the total profit number for the last 30 days for all products.

Dashboard -> Profit widget -> Redirecting to the Product info page and back

Redirecting to the Product info page from "Top 5 products" tab on the "Profit" widget

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted

Pre-conditions

Log in as user/admin/RA to the connected account.

There are more then 5 products with matches in the account.

Products have filled "Cost price" attribute.

Account has products with minimum 5 Brands.

On the Datafeeds page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

On the Settings page Google Analytics settings-> Data interval is 30 days.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	On the "Profit" widget click on the "Top 5 products" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 products" tab display the numbers of profit for the last 30 days for the top 5 products with the h ighest profit to the lowest.
Step	2
Action	Pay attention to the "A" product profit number. Opposite the "A" product click on the red "Right" arrow.
Input data	
Expected result	Redirecting to the "A" Product info page.
Step	3
Action	Click on the "Back to dashboard" button.
Input data	
Expected result	Dashboard page is opened. "Profit" widget opened on the "Total" tab, profit for the last 30 days.

Dashboard -> Profit widget -> Redirecting to the Product list page

Redirecting to the Product list page from "Top 5 brands" tab on the "Profit" widget.

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
_	Manual	Actual	No

Pre-conditions

Log in as user/admin/RA to the connected account.

There are more then 5 products with matches in the account.

Products have filled "Cost price" attribute.

Account has products with minimum 5 Brands.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

On the Settings page Google Analytics settings-> Data interval is 30 days.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	On the "Profit" widget click on the "Top 5 brands" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 brands" tab display the numbers of profit for the last 30 days for the top 5 brands with the high est profit to the lowest.
Step	2
Action	Pay attention to the "A" brand profit number. Opposite the "A" brand click on the red "Right" arrow.
Input data	
Expected result	Redirecting to the Product list page filtered by "A" brand.
Step	3
Action	Click on the "Filters" button.
Input data	
Expected result	On the "Filter selection" "A" brand is represented.

Dashboard -> Profit widget -> Displaying the "Profit" widget when GA is disabled (Admin)

Check displaying the "Profit" widget when GA is disabled on the Datafeeds page

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin to the connected account.

Products have filled "Cost price" attribute.

On the Datafeeds page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	On the Dashboard page pay attention to the "Profit" widget, click throught 3 radio buttons.
Input data	
Expected result	The "Profit" widget displays tab's relevant data.
Step	2
Action	Open the Datafeeds page click on "G" Google Analitycs sign.
Input data	
Expected result	"Feed's name Google Analytics" popup was displayed.
Step	3
Action	Turn off "Enable Google Analytics" toogle.
Action Input data	Turn off "Enable Google Analytics" toogle.
Input data	Turn off "Enable Google Analytics" toogle. The "Enable Google Analytics" toogle is off.
Input data	
Input data Expected result	The "Enable Google Analytics" toogle is off.
Input data Expected result Step	The "Enable Google Analytics" toogle is off.

Step	5
Action	Open the Dashboard page.
Input data	
Expected result	Dashboard page is opened. Pay attention that "Profit" widget is not displayed anymore.

Dashboard -> Settings -> "Calculate gross profit with VAT"-> ON->Off -> Profit widget (Admin)

Displaying "Profit" widget data according to the ON/OFF "Calculate gross profit with tax" settings

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as an admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

On the "Settings" page Option "Calculate gross profit with tax" is on.

"Profit" widget is added.

Step	1
Action	On the Dashboard page pay attention to the "Profit" widget's tabs, differend data.
Input data	
Expected result	Profit data for the single purchase is calculated by formula including VAT % Profit= ((display price/((100+VAT)/100)) - cost price).
Step	2
Action	Open the "Settings" page.
Input data	
Expected result	The "Settings" page is opened.
Step	3
Action	On the "Options" section turn off the "Calculate gross profit with tax" toogle.
Input data	
Expected result	The toogle is off next to the "Calculate gross profit with tax" option.
Step	4
Action	Click on the "Save" button.

Expected result	"Account settings updated" and "Settings saved" information messages was shown.
Step	5
Action	Open the Dashboard page, pay attention to the "Profit" widget's tabs, differend data.
Input data	
Expected result	Profit data for the single purchase is calculated by formula without VAT % Profit= Display price-Cost price Formulas are here https://docs.google.com/spreadsheets/d/1joOdP5pPIFdH7kbHmUnnBLbYub8m1TzKioHrnyOrq5O/edit#gid=1235726146

Input data

Dashboard -> Profit widget -> Check data in widget

Проверить подсчет профита за выбранный период и сравнить его с базой. Так же проверить топ 5 продуктов и то 5 брендов и сравнить с базой

Severity	Priority	Behavior	Туре
O Normal	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)	Manual	Actual	No

1

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. В аккаунте подключена гугл аналитика, собраны данные о продуктах, у продуктов есть значения "cost price".
- 3. В настройках гугл аналитики в "Data interval "выбрано "1month".
- 4. Добавить "Profit" виджет на Dashboard страницу.

Step	1
Action	Перейти на Dashboard страницу.
Input data	
Expecte result	d 1. В "Profit" виджете отображается "Last 30 days" период и активен "Total" радиобаттон.
Step	2
Action	Сравнить данные с "Profit" виджета и с базы данных.
Input data	SELECT SUM(google_analytics_data->>'\$."30"."profit"') as total_profit FROM sphinx_products WHERE account_id = account_id LIMIT 0, 10
Expecte result	d 1. Результаты в базе данных соотвествуют отображению в "Profit" виджете.
Step	3
Action	Изменить период на "Last 7 days" и сравнить данные с виджета и с базы данных.
Input data	SELECT SUM(google_analytics_data->>'\$."7"."profit"') as total_profit FROM sphinx_products WHERE account_id = 10827 LIMIT 0, 10
Expecte result	d. Результаты в базе данных соотвествуют отображению в "Profit" виджете.
Step	4
Action	Нажать на "Top5 products" радиобаттон и сравнить данные с виджета и с базы данных.

Input data

SELECT id, title, google_analytics_data->>'\$."7"."profit"' as profit FROM sphinx_products WHERE account_id = account_id HAVING profit != 0 ORDER BY profit DESC LIMIT 0, 5

Expected Pesyльтаты в базе данных соотвествуют отображению в "Profit" виджете.

Step 5

Action Нажать на "Top5 brands" радиобаттон и сравнить данные с виджета и с базы данных.

data nds_profit DESC LIMIT 0, 5

Expected Peзультаты в базе данных соотвествуют отображению в "Profit" виджете.

Step 6

Action Отфильтровать виджет по любому бренду и сравнить данные общего профита за 30 дней с виджета и с базы данных.

SELECT brand_id, SUM(google_analytics_data->>'\$."7"."profit"') as brands_profit FROM sphinx_products WHERE account_id = account_id <u>AND</u> google_analytics_data->>'\$."7"."profit"' <u>IS NOT</u> NULL <u>AND</u> google_analytics_data->>'\$."7"."profit"' != 0 GROUP BY brand_id ORDER BY brand_id OR

data

SELECT SUM(google_analytics_data->>'\$."30"."profit"') as total_profit FROM sphinx_products WHERE account_id = account_id AND brand_id

d IN (brand_id) LIMIT 0, 10

 $\mathsf{Expected}_1$. Результаты в базе данных соотвествуют отображению в "Profit" виджете. result

Input data

Input

E2E: Dashboard -> Profit widget -> Add

Adding "Profit" widget to the Dashboard page by the "+ New widget" button.

"Profit" widget displays total product profit, single product profit, and product profit by brand over a specified period of time (last day, 7 days, 14 days, 30 days, 6 months).

Severity	Priority	Behavior	Туре	
☆ Critical	↑ High	Positive	Regression	
Milestone	Automation	Status	Is Muted	
-	Manual	Actual	No	
2	2	3		

Pre-conditions

- 1. There are products with matches in the account.
- 2. On the Datafeeds page Google analytics is enabled, data is connected or generated test data.
- 3. On the Settings page Google Analytics widgets is enabled.
- 4. Dashboard page is opened.

Step	1
Action	Click the "Gear" icon at the header of the page.
Input data	
Expected result	 All the menus are become not active. The buttons "Save", "+ New widget", and the "+ New widget" card was displayed.
Step	2
Action	Click the "+ New widget" button.
Input data	
Expected result	The "Add widgets" popup was opened.
Step	3
Action	Add "Profit" widget by clicking the "+" icon to the left of a widget name.
Input data	
Expected result	Counts of the "Profit" widget was increasing on 1.
Step	4
Action	Click the "Save" button.
Input data	
Expected result	 The "Add widget" popup was closed and the "Profit" widget is displayed in its default state. The "Widgets were successfully created" message was displayed.

E2E: Dashboard -> Profit widget -> Filters-> "Brands"

"Profit" widget displaying for selected Brand

SeverityPriorityBehaviorType- Not setO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Account has products from minimum 3 different brands.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

Dashboard page is opened.

"Profit" widget is added.

Steps to reproduce

Action

Click on the "Save" button.

Step	1
Action	On the "Profit" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Brands" search field.
Input data	
Expected result	Dropdown with Brands was displayed.
Step	3
Action	Select 2 Brands, "Test1" and "Test2", by clicking on checkboxes.
Input data	
Expected result	Selected 2 Brand's checkboxes are filled with checkmarks.
Step	4

Input data	
Expecte result	ed Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.
Step	5
Action	Click on the "Top 5 brands" radio button. Compare the profit number for this Brands on the "Profit" widget with the "Total profit" parameter on the "Goog le analytics" widget for the specific period filtered by these Brands.
Input data	
Expecte result	The "Profit" widget display was changed. On the "Top 5 brands" list there are only "Test1" and "Test2" Brands are represented. The Profit number on the "Profit" widget is the same as "Total profit" parameter on the "Google analytics".
Step	6
Action	Click on the "Top 5 products" radio button.
Input data	
Expecte result	ed The "Profit" widget display was changed. On the "Top 5 products" list there are only products from "Test1" and "Test2" Brands are represented.

E2E: Dashboard -> Profit widget -> Filters-> "Product Types"

Profit widget displaying for selected Product type

SeverityPriorityBehaviorType- Not setO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
Products have filled "Cost price" attribute.
Account has products with minimum 3 different Product types.
On the Datafeed page Google analytics is enabled, data is connected or generated test data.
On the Settings page Google Analytics widgets is enabled.
Dashboard page is opened.
"Profit " widget is added.

1
On the "Profit" widget click on the red editing "Pen".
"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
2
Click on "Product types" search field.
Dropdown with Product types was displayed.
3
Select "Test1" Product type by clicking on checkbox.
Selected "Test1" Product type checkbox is filled with the checkmark.
4
Click on the "Save" button.

Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. "Profit" widget is opened on the "Total" tab.
Step	5
Action	Click on the "Top 5 products" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 products" list there are only products with "Test1" Product type.
Step	6
Action	Opposite "A" product click on the red "Arrow".
Input data	
Expected result	Redirecting to the "A" Product info page. Pay attention to the "A" product Product type.

E2E: Dashboard -> Profit widget -> Filters-> "Tags"

Profit widget displaying for selected Tags

Behavior Severity **Priority Type** - Not set O Medium Not set Other Is Muted Milestone **Automation Status** Manual Actual No

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Products have minimum 3 different Tags.

"A" product has "Test1" Tag.
"B" product has "Test2" Tag.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	On the "Profit" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expected result	Dropdown with Tags was displayed.
Step	3
Action	Select "Test1" Tag by clicking on checkbox. Write to the Tag's search field "Test2" tag's name. Select "Test2" Tag by clicking on checkbox.
Input data	
Expected result	Selected 2 Tag's checkboxes are filled with checkmarks.
Step	4
Action	Click on the "Save" button.

Input data	
Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. The "Profit" widget is opened on the "Total" tab.
Step	5
Action	Click on the "Top 5 products" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 products" list there are only products with "Test1" and "Test2" Tags.
Step	6
Action	Opposite "A" product click on the red "Arrow".
Input data	
Expected result	Redirecting to the "A" Product info page. Pay attention to the "A" applied tags ("Test1").
Step	7
Action	Click on the "Back to Dashboard" button.
Input data	
Expected result	Redirecting to the Dashboard page. The "Profit" widget is still filtered by "Test1" and "Test2" Tags, set on the "Total" radio button.
Step	8
Action	Click on the "Top 5 products" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 products" list there are only products with "Test1" and "Test2" Tags.

Step

Action

Input data

9

Opposite "B" product click on the red "Arrow".

Expected result Redirecting to the "B" Product info page. Pay attention to the "B" applied tags ("Test12).

E2E: Dashboard -> Profit widget -> Filters-> "Dynamic tags"

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account. Products have filled "Cost price" attribute.

Products have minimum 2 different Dynamic tags.

"A" product has "Test1" Dynamic tags.

"B" product has "Test2" Dynamic tags.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

Dashboard page is opened.

"Profit " widget is added.

Step	1
Action	Click on the red pen "Edit" button on "Profit " widget.
Input data	
Expected result	"Choose what you want to see" popup is displayed.
Step	2
Action	Click on the "Tags" field and select the dynamic tags "Test1", "Test2" (or write the dynamic tag name in the s earch field and select it by check).
Input data	
Expected result	The dynamic tags name is displayed in the "Tags" search field.
Step	3
Action	Click on the "Save" button.
Input data	
Expected result	 Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. The "Profit" widget is opened on the "Total" tab

Action Click on the "Top 5 products" radio button. Input data Expected 1. The "Profit" widget display was changed. 2. On the "Top 5 products" list there are only products with "Test1" and "Test2"dynamic tags. result Step 5 Action Opposite "A" product click on the red "Arrow". Input data **Expected** 1. Redirecting to the "A" Product info page. result 2. Pay attention to the "A" applied dynamic tags ("Test1"). Step 6 Action Click on the "Back to Dashboard" button. Input data Expected 1. Redirecting to the Dashboard page. 2. The "Profit" widget is still filtered by "Test1" and "Test2" dynamic tags, set on the "Total" radio button. result 7 Step Click on the "Top 5 products" radio button. Action Input data Expected 1. The "Profit" widget display was changed. 2. On the "Top 5 products" list there are only products with "Test1" and "Test2" dynamic tags. result Step 8 Opposite "B" product click on the red "Arrow". Action Input data

1. Redirecting to the "B" Product info page.

2. Pay attention to the "B" applied dynamic tags ("Test12)

Expected

result

E2E: Dashboard -> Profit widget -> Filters -> "Imported tags"

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported_tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported_tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Products have minimum 3 different Imported Tags.

- "A" product has "Test1" Imported Tag.
 "B" product has "Test2" ImportedT ag.
 On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 7. On the Settings page Google Analytics widgets is enabled.
- 8. On the Settings page "Calculate gross profit with VAT" options is enabled.
- 9. Dashboard page is opened.
- 10. "Profit " widget is added.

Steps to reproduce

Step

Action	On the "Profit" widget click on the red editing "Pen".
Input data	
Expecte result	ed _" EDIT PROFIT WIDGET" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expecte result	ed Dropdown with Tags was displayed.
Step	3
Action	Click on "Imported Tags" field.
Input data	

Expecte result	ed Dropdown with Imported Tags was displayed.
Step	4
Action	Select "Test1" Tag by clicking on checkbox.
Input data	
Expecte result	edSelected imported tag checkbox are filled with checkmark.
Step	5
Action	Click on the "Save" button.
Input data	
Expecte result	1. Widgets editing mode was closed. 2. Information message with "Widget was successfully updated" was shown. 3. The "Profit" widget is opened on the "Total" tab. 4. The "Profit" widget is filtered by "Test1" Imported Tag.
Step	6
Action	Run query in DB
Input data	SELECT sum(google_analytics_data->>'\$." X "."profit_wi thout_vat"') as PROFIT FROM sphinx_products inner join product_imported_tags on sphinx_products.id = product_imported_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id where sphinx_products.account_id = account_id and imported_tags.name = ' imported_tags_name ';
Expecte result	ed Value in the same with "Total" tab.
Step	7
Action	Change tab on "Top 5 products" and run query in DB:
Input data	SELECT title, google_analytics_data->>'\$." X "."profit_w ithout_vat"' as PROFIT FROM sphinx_products inner join product_imported_tags on sphinx_products.id = product_imported_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id where sphinx_products.account_id = account_id and imported_tags.name = ' imported_tags_name ' ORDE R BY `PROFIT` ASC
Expecte result	ed Value in the same with "Top 5 products" tab.
Step	8
Action	Value in the same with "Top 5 products" tab.

Input data SELECT brands.name AS BRAND, sum(google_analytics_data->>'\$."X"."profit_without_vat"') as PROFIT FROM brands inner join sphinx_products on brands.id = sphinx_products.brand_id inner join product_imported_tags on sphinx_products.id = product_imported_tags.product_id inner join imported_tags on product_imported_tags.imported_tag_id = imported_tags.id where sphinx_products. account_id = account_id and imported_tags.name = '< span style="color:var(--md-font-color-failed)">imported_tags_name' GROUP BY brands.name ORDER BY `PROFIT` DESC

Expected Value in the same with "Top 5 brands" tab. result

E2E: Dashboard -> Profit widget -> Filters-> "Labels"

Profit widget displaying for selected Labels

Severity Priority Behavior Type
O Normal O Medium Not set Other

Milestone Automation Status Is Muted
- Manual Actual No

3

Pre-conditions

Log in as user/admin/RA to the connected account.
There are products with matches in the account.
Products have filled "Cost price" attribute.
Account has products with minimum 2 different Product labels.
On the Datafeed page Google analytics is enabled, data is connected or generated test data.
On the Settings page Google Analytics widgets is enabled.
Dashboard page is opened.
"Profit" widget is added.

1
On the "Profit" widget click on the red editing "Pen".
"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
2
Click on "Labels" search field.
Dropdown with Labels was displayed.
3
Select "Test1" Label by clicking on checkbox.
Selected "Test1" Label checkbox is filled with the checkmark.
4
Click on the "Save" button.

Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.
Step	5
Action	Click on the "Top 5 brands" radio button.
Input data	
Expected result	The "Profit" widget display was changed. On the "Top 5 brands" list there are only products with "Test1" Label.
Step	6
Action	Opposite "A" brand click on the red "Arrow".
Input data	
Expected result	Redirecting to the Product list with products that belong to the "Test1" Label.
Step	7
Action	Click on the any product card.
Input data	
Expected result	Redirecting to the selected product's Product info page. Pay attention to the Labels attribute. The product has "Test1" Label.

E2E: Dashboard -> Profit widget -> Filters-> "Reset to default"

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Functional
Milestone	Automation	Status	Is Muted
_	Manual	Actual	No

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled.
- 5. Dashboard page is opened.
- 6. "Profit" widget is added and filtered by custom parameters (by Brands, Tags, Dynamic tags, Product types, Labels).

Step	1
Action	Run a query in the database
Input data	SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
	In the fields params displayed JSON which displayed your filters: The result in the database is the same as on the page {
Expected result	<pre>"origin": null, "brandslds": [brand_id_1], "labelslds": [label_id_1], "dynamicTags": [dynamic_tag_id_1], "customLabels": ["name"], "productTypesIds": [product_type_id_1] }</pre>
Step	2
Action	Click on the red pen "Edit" button in the upper left corner on the "Profit" widget.
Input data	
Expected result	"Choose what you want to see" pop-up is displayed.
Step	3
Action	Click on "Reset to default" button.
Input data	

Expected result	"Widget was successfully updated" information message was displayed.	
Step	4	
Action	Click on the red cross in the top right corner.	
Input data		
Expected result	"Profit" widget is displaying in the defoult state, on the "Total" radio button, for all products, data interval ma tches the settings.	
Step	5	
Action	Run a query in the database	
Input data	SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget	
Expected result	In the fields params displayed JSON which displayed your filters: The result in the database is the same as on the page { "origin": null, "brandslds": [], "labelslds": [], "dynamicTags": [], "customLabels": [], "productTypeslds": [] }	

Dashboard -> Conversion rate -> "Last N days" dropdown (Admin)

"Conversion rate" widget displaying for selected period on the Settings page

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 5. "Conversion rate" widget is added.

Steps to reproduce

Action

Step	1	
Action	On the "Conversion rate" pay attention to the "Last 30 days" dropdown. Click on the "Last 14 days".	
Input data		
Expected result	"Conversion rate" widget display was changed, data for 14 last days period is representing now.	
	SES CONVESSION RATE Lust 70 days. Lust 70 day The Just 70 day	
Step	2	
Action	On the "Settings" page, "Google Analytics" sector choose "Data interval"-> 6 months.	
Input data		
Expected result	"Data interval" "6 months" radio button is choosed.	
Step	3	
Action	On the "Settings" page click on the "Save" button.	
Input data		
Expected result	"Account settings updated" and "Settings saved" informational message was displayed.	
Step	4	

On the Dashboard page, "Conversion rate" widget pay attention to the period displaying.

Input data

Expected result Period displaying is "Last 6 months".



Dashboard -> Conversion rate -> Filters -> "Sort by" dropdown

Widget display according to high and low scores by Visitors, Revenue, Number of Sales, Conversion rate

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 5. "Conversion rate" widget is added.

Step	1
Action	Click on the "Revenue Highest" dropdown.
Input data	
Expected result	Dropdown with the "Visitors", "Revenue", "Number of Sales", "Conversion rate" options was opened. "Revenue" option is highlighted, displayed in red letters on a white background.
Step	2
Action	Hover over "Visitors" option. Click on the "Lowest".
Input data	
Expected result	Next to the "Visitors" option "Lowest/Highest" menu was opened. After loader widget's displaying was changed. "Visitors Lowest" dropdown is displaed. Top 5 products with the lowest number of visitors for the last 30 days was displayed. Pay attention to the "Visitors" column, displays from smallest to largest number.
Step	3
Action	Click on the "Visitors Lowest" dropdown.
Input data	
Expected result	Dropdown with the "Visitors", "Revenue", "Number of Sales", "Conversion rate" options was opened. "Visitors" option is highlighted, displayed in red letters on a white background.
Step	4
Action	Hover over "Number of Sales" option. Click on the "Highest".
Input data	

Next to the "Number of Sales" option "Lowest/Highest" menu was opened.

After loader widget's displaying was changed.

Expected result "Number of Sales Highest" dropdown is displaed.

Top 5 products with the highest number of sales for the last 30 days was displayed.

Pay attention to the "Number of Sales" column, displays from largest to smallest number.

5 Step

Action Click on the "Number of Sales Highest" dropdown.

Input data

Expected result Dropdown with the "Visitors", "Revenue", "Number of Sales", "Conversion rate" options was opened. "Number of Sales" option is highlighted, displayed in red letters on a white background.

6 Step

Action Hover over "Conversion rate" option. Click on the "Lowest".

Input data

Next to the "Conversion rate" option "Lowest/Highest" menu was opened.

After loader widget's displaying was changed.

Expected result "Conversion rate Lowest" dropdown is displaed.

Top 5 products with the lowest "Conversion rate" number for the last 30 days was displayed.

Pay attention to the "Conversion rate" column, displays from smallest to largest number.

Dashboard -> Conversion rate -> Redirecting to the Product info page and back with filters

Severity	Priority	Behavior	Type
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Account has minimum 2 different product types, 2 different brands.
- 4. "A" product has "Test" product type and belongs to the "Brand1" brand, has highest "Visitors" rate.
- 5. On the Datafeeds page Google analytics is enabled, data is connected or generated test data.
- 6. On the Settings Data interval = 1 month.7. "Conversion rate" widget is added and filtered by "Test" product type and "Brand1"brand.

Step	1
Action	Open Dashboard, pay attention to the "Conversion rate" widget.
Input data	
Expected result	"Conversion rate" widget displays Top 5 products with the highest revenue for the last 30 days from "Test" product type and "Brand1" brand.
Step	2
Action	Click on the "Revenue: Highest" dropdown. Hover to the "Visitors" click on "Visitors->highest" option.
Input data	
Expected result	 Dropdown with: "Visitors", "Revenue", "Number of sales", "Conversion rate" is opened. The "Conversion rate" changed for the "Visitors->highest" priority. The "A" product is on the top of the list.
	ASS CONVESSION RATE Last 32 day - Backet right his backetoper 1 six Backets backetoper 2 day bone 500 for branch — Backets backetoper 2 day bone 500 for branch — Backets backetoper 2 days 1 day bone 500 for 1 da back Dopper College 1 days 1 days backet 1 days b
Step	3
Action	Opposite the "A" product click on the red arrow "Right".
Input data	
Expected result	 Redirecting to the "A" product info page. Pay attention to the product title, brand, product type and "Google analytics" data. Compare it with the same period.

Step	4
Action	Click on the "Back to dashboard" button.
Input data	
Expected result	 Dashboard page is opened. "Conversion rate" widget displays Top 5 products with the highest revenue for the last 30 days from "Test" p roduct type and "Brand1" brand.

Dashboard -> Displaying the "Conversion rate" widget when GA is disabled (Admin)

Check displaying the "Conversion rate" widget when GA is disabled on the Datafeeds page

Severity	Priority	Behavior	Туре
^ Major	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
	Manual	Actual	No

Pre-conditions

- 1. Products have filled "Cost price" attribute.
- 2. On the Datafeeds page Google analytics is enabled, data is connected or generated test data.
- 3. On the Settings page Google Analytics widgets is enabled.
- 4. Dashboard page is opened.
- 5. "Conversion rate" widget is added.

Step	1
Action	On the Dashboard page pay attention to the "Conversion rate" widget.
Input data	
Expected result	1. The "Conversion rate" widget displays relevant data.
Step	2
Action	Open the Datafeeds page click on "G" Google Analitycs sign.
Input data	
Expected result	1. "Feed's name Google Analytics" popup was displayed.
Step	3
Action	Turn off "Enable Google Analytics" toogle.
Input data	
Expected result	1. The "Enable Google Analytics" toogle is off.
Step	4
Action	Click on the "Save " button.
Input data	
Expected result	1. "Google Analytics settings saved" informational message was shown.

Step	5
Action	Open the Dashboard page.
Input data	
Expected result	 Dashboard page is opened. Pay attention that "Conversion rate" widget is not displayed anymore.

Dashboard -> Conversion rate widget -> Check data in widget

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Not set	Regression
Milestone			
Milestone	Automation	Status	Is Muted

1

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. Собрана Гугл аналитика, у продуктов есть значения у атрибута "cost price".
- 3. B Settings-> GOOGLE ANALYTICS-> Data interval выбрано "1month.
- 4. "Conversion rate" виджет добавлен на Dashboard в дефолтном, не отфильтрованном состоянии.
- 5. Открыть Dashboard.

Step	1
Action	Обратить внимание на "Conversion rate" виджет.
Input data	
Expecte result	d 1. Отображается период "Last 30 days", дефолтно "Revenue Highest".
Step	2
Action	Проверить значения "Revenue Highest" в базе.
Input data	SELECT id, title, CAST(google_analytics_data->>'\$."30"."sales"' AS DECIMAL(7,2)) AS Revenue FROM sphinx_products WHERE google_analytics_data-
Expecte result	d _{1. Значения} сходятся.
Step	3
Action	Нажав на дропдаун "Revenue Highest" выбрать "Revenue Lowest". Проверить значения "Revenue Lowest" в БД.
Input data	SELECT id, title, CAST(google_analytics_data->>'\$."30"."sales"' AS DECIMAL(7,2)) AS Revenue FROM sphinx_products WHERE google_analytics_data-
Expecte result	1. Подборка продуктов на виджете изменилась. 2. Отображаются продукты с наименьшим "Revenue". 3. Значения сходятся.
Step	4
Action	Нажав на дропдаун "Revenue Lowest" выбрать "Conversion rate Highest". Обратить внимание на "Conversion rate Highest", проверить значения в БД.

Input data

SELECT id, title, CAST(google_analytics_data->>'\$."30"."conversion_rate"' AS DECIMAL(5,2)) AS conversion_rate FROM sphinx_products WHERE go

Expected. Подборка продуктов на виджете изменилась. Отображаются продукты с наибольшим "Conversion rate".

result

3. Значения сходятся.

Step 5

Action

Нажав на дропдаун "Conversion rate Highest" выбрать "Conversion rate Lowest". Обратить внимание на "Conversion rate Lowest", проверить значения в БД.

SELECT id, title, CAST(google_analytics_data->>'\$."30"."conversion_rate"' AS DECIMAL(5,2)) AS conversion_rate

FROM sphinx_products

Input data

WHERE google_analytics_data->>'\$."30"."conversion_rate"' IS NOT NULL

AND account_id = account_id

AND CAST(google_analytics_data->>'\$."30"."conversion_rate"' AS DECIMAL(5,2)) > 0 ORDER BY CAST(google_analytics_data->>'\$."30"."conversion_rate"' AS DECIMAL(5,2)) ASC

LIMIT 5:

Expected 1. Подборка продуктов на виджете изменилась. 2. Отображаются продукты с наименьшим "Conversion rate". result

3. Значения сходятся.

Step 6

Action

Нажав на дропдаун "Conversion rate Lowest" выбрать "Visitors Highest". Обратить внимание на "Visitors Highest", проверить значения в БД.

Input data

result

SELECT id, title, CAST(google_analytics_data->>'\$."30"."visitors"' AS DECIMAL) AS visitors FROM sphinx_products WHERE google_analytics_data->

Expected 1. Подборка продуктов на виджете изменилась.

Отображаются продукты с наибольшим количеством"Visitors".

3. Значения сходятся.

7 Step

Action

Нажав на дропдаун "Visitors Highest" выбрать "Visitors Lowest". Обратить внимание на "Visitors Lowest", проверить значения в БД.

Input data

SELECT id, title, CAST(google_analytics_data->>'\$."30"."visitors"' AS DECIMAL) AS visitors FROM sphinx_products WHERE google_analytics_data->

Expected 1. Подборка продуктов на виджете изменилась.
2. Отображаются продукты с наименьшим количеством"Visitors".

result

3. Значения сходятся.

8 Step

Action

Нажав на дропдаун "Visitors Lowest" выбрать "Number of Sales Highest".

Обратить внимание на "Sales", проверить значения в БД.

Input data

SELECT id, title, CAST(google_analytics_data->>'\$."30"."quantity"'AS DECIMAL) AS Sales FROM sphinx_products WHERE google_analytics_data->>

Expected. Подборка продуктов на виджете изменилась. 2. Отображаются продукты с наибольшим количеством"Sales".

result 3. Значения сходятся.

Нажав на дропдаун "Number of Sales Highest" выбрать "Number of Sales Lowest". Action Обратить внимание на "Sales", проверить значения в БД. Input SELECT id, title, CAST(google_analytics_data->>'\$."30"."quantity"'AS DECIMAL) AS Sales FROM sphinx_products WHERE google_analytics_data->> data Expected 1. Подборка продуктов на виджете изменилась. result 2. Отображаются продукты с наименьшим количеством"Sales". result 3. Значения сходятся. 10 Step Изменить на виджете период времени (например "Last 6 months"). Отфильтровать виджет (например по бренду). Action Сравнить значения для "Conversion rate Highest" в БД. Input

Expected 1. Подборка продуктов на виджете изменилась. 2. Отображаются продукты с наибольшим "Conversion rate" выбранного бренда за период 180 дней.

SELECT id, title, CAST(google_analytics_data->>'\$."180"."conversion_rate" AS DECIMAL(5,2)) AS conversion_rate FROM sphinx_products WHERE

result 3. Значения сходятся.

data

E2E: Dashboard -> Conversion rate -> Filters -> "Brands"

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin to the connected account.
There are products with matches in the account.
Products have filled "Cost price" attribute.
Account has products from minimum 2 brands.
On the Datafeed page Google analytics is enabled, data is connected or generated test data.
On the Settings page Google Analytics widgets is enabled.
"Conversion rate" widget is added.

Step	1
Action	On the "Conversion rate" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Brands" search field.
Input data	
Expected result	Dropdown with Brands was displayed.
Step	3
Action	Select "Test1" brand by clicking on checkbox.
Input data	
Expected result	Selected Brand checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	

Expected result

Widgets editing mode was closed. Information message with "Widget was successfully updated" was show n.

Top 5 products with the highest revenue from "Test1" Brand are represented.

E2E: Dashboard -> Conversion rate -> Filters -> "Product types"

"Conversion rate" widget displaying for selected Product type

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
	Automation	Status	is muteu

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Account has products from minimum 2 different product types.
- 4. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 5. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 6. "Conversion rate" widget is added.

Step	1
Action	On the "Conversion rate" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Product types" search field.
Input data	
Expected result	Dropdown with product types was displayed.
Step	3
Action	Enter "Test1" product type's name into the search field.
Input data	
Expected result	 Search field is filled with "Test1". On the dropdown list is the "Test1"product type.
Step	4
Action	Select "Test1" product type by clicking on checkbox.
Input data	

Expected result	Selected Product type checkbox is filled with the checkmark.
Step	5
Action	Click on the "Save" button.
Input data	
Expected result	 Widgets editing mode was closed. Information message with "Widget was successfully updated" was sho wn. Top 5 products with the highest revenue for the last 30 days from "Test1" product type are represented.

E2E: Dashboard -> Conversion rate -> Filters -> "Tags"

"Conversion rate" widget displaying for selected Tags

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Minimum 2 different tags are added for products in the account.
- 4. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 5. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 6. "Conversion rate" widget is added.

Step	1
Action	On the "Conversion rate" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expected result	Dropdown with tags was displayed.
Step	3
Action	Select "Test1" tag by clicking on checkbox.
Input data	
Expected result	Selected tag's checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	

Expected result

Widgets editing mode was closed. Information message with "Widget was successfully updated" was show n.
Top 5 products with the highest revenue for the last 30 days from "Test1" tag are represented.

E2E: Dashboard -> Conversion rate -> Filters -> "Dynamic tags"

"Conversion rate" widget displaying for selected Dynamic tags

SeverityPriorityBehaviorTypeO NormalO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Minimum 2 different dynamic tags are added for products in the account.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.

"Conversion rate" widget is added.

Step	1
Action	On the "Conversion rate" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	.Click on "Tags" search field
Input data	
Expected result	Dropdown with tags was displayed.
Step	3
Action	Select "Test1" dynamic tag by clicking on checkbox.
Input data	
Expected result	Selected dynamic tag checkbox is filled with the checkmark.
Step	4
Action	Click on the "Save" button.
Input data	

Expected result

Widgets editing mode was closed. Information message with "Widget was successfully updated" was show n.
Top 5 products with the highest revenue for the last 30 days from "Test1" dynamic tag are represented.

E2E: Dashboard -> Conversion rate widget -> Filters -> "Imported tags"

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	3

Pre-conditions

- 1. Log in as user/admin to the connected account.
- 2. There are products with matches in the account.3. Products have filled "Cost price" attribute.
- 4. Minimum 2 different imported tags are added for products in the account.
- 5. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 6. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 7. "Conversion rate" widget is added.

Step	1	
Action	On the "Conversion rate" widget click on the red editing "Pen".	
Input data		
Expected _{"EDIT} CONVERSION RATE WIDGET" filter's popup was displayed. result		
Step	2	
Action	Click on "Tags" search field.	
Input data		
Expected Dropdown with tags was displayed. result		
Step	3	
Action	Click on "Imported Tags" field.	
Input data		
Expected Dropdown with Imported Tags are opened. result		
Step	4	

Action Select "ImportedTag1" imported tag by clicking on checkbox. Input data $\label{eq:selected} \textbf{Expected}_{\textbf{Selected Imported tag checkbox is filled with the checkmark}.$ result Step Click on the "Save" button. Action Input data Expected. Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. 2. Top 5 products with the highest revenue for the last 30 days from "ImportedTag1" imported tag are represente result Step 6 Action Run query in DB: SELECT title as TITLE, google_analytics_data->>'\$."X". "visitors"' AS Visitors, google_analytics_data->>'\$."X". "sales" as Revenue, google_analytics_data->>'\$."X"." quantity" AS Sales ,google analytics data->>'\$."X"."c Input onversion rate" as Conversion rate FROM sphinx products inner join product imported tags on sphinx product data s.id = product imported tags.product id inner join imported tags on product imported tags.imported tag id = imported tags.id where sphinx products.account id = account id and imported tags name = 'imported tags name </ span>' ORDER BY `Revenue` DESC

 $\label{eq:conversion} \textbf{Expected}_{\text{Values in the same with "Conversion rate" widget.}}$

result

E2E: Dashboard -> Conversion rate -> Filters -> "Labels"

"Conversion rate" widget displaying for selected Label

SeverityPriorityBehaviorType- Not setO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Account has products from minimum 2 different product labels (added in the feed).

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.

"Conversion rate" widget is added.

Step	1	
Action	On the "Conversion rate" widget click on the red editing "Pen".	
Input data		
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.	
Step	2	
Action	Click on "Labels" search field.	
Input data		
Expected result	Dropdown with labels was displayed.	
Step	3	
Action	Enter "Test1" product label's name into the search field.	
Input data		
Expected result	Search field is filled with "Test1". On the dropdown list represented the "Test1" label.	
Step	4	
Action	Select "Test1" label by clicking on checkbox.	
Input data		

Expected result	Selected label's checkbox is filled with the checkmark.	
Step	5	
Action	Click on the "Save" button.	
Input data		
Expected result	Widgets editing mode was closed. Information message with "Widget was successfully updated" was show n. Top 5 products with the highest revenue for the last 30 days from "Test1" label are represented.	

E2E: Dashboard -> Conversion rate -> Filters -> "Reset to default"

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled.
- 5. Dashboard page is opened.
- 6. "Conversion rate" widget is added and filtered by custom parameters (by Brands, Tags, Dynamic tags, Product types, Labels).

```
Step
              1
Action
              Run a query in the database
Input data SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
              In the fields params displayed JSON which displayed your filters:
              The result in the database is the same as on the page
              "origin": null,
              "brandslds": [
              brand_id_1
              "labelsIds": [
              label_id_1
Expected
result
              "dynamicTags": [
              dynamic_tag_id_1
               "customLabels": [
              "name"
              "productTypesIds": [
              product_type_id_1
              2
Step
Action
              Click on the red pen "Edit" button in the upper left corner on the "Conversion rate" widget.
Input data
Expected
              "Choose what you want to see" pop-up is displayed.
result
Step
              3
Action
              Click on "Reset to default" button.
```

Input data	
Expected result	"Widget was successfully updated" information message was displayed.
Ston	4
Step	*
Action	Click on the red cross in the top right corner.
Input data	
Expected result	"Conversion rate" widget is displaying in the defoult state, top 5 products with the highest revenue to lowest, d ata interval matches the settings.
Step	5
Step Action	5 Run a query in the database
Action	
Action	Run a query in the database

Dashboard -> Google Analytics widget -> Filters-> "Tags"

"Google Analytics" widget displaying for selected Tags

SeverityPriorityBehaviorTypeO NormalO MediumNot setOtherMilestoneAutomationStatusIs Muted-Should be automatedActualNo

2

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Account has products with minimum 2 different Tags.
- 4. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 5. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 6. "Google Analytics" widget is added.

Step	1
Action	On the "Google Analytics" widget click on the red editing "Pen".
Input data	
Expecte result	CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expecte result	Dropdown with tags was displayed.
Step	3
Action	Enter "Test1" Tag's name into the search field.
Input data	
Expecte result	ed. Search fileld is filled with "Test1". 2. On the dropdown list is the "Test1"tag.
Step	4
Action	Select "Test1" tag by clicking on checkbox.



Expected

result Selected tag's checkbox is filled with the checkmark.

Step 5

Action Click on the "Save" button.

Input data

Expected. Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. result 2. On the charts there are only information and data for all products from "Test1"tag for the last 30 days.

Step 6

Сравнить данные с запросами в базе.

Action

- 1. По датам и точкам.
- 2. Общие данные (сумма).

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchase s) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversi on_rate, google_analytics.analytics_date from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date e` >= '2022-12-21' and `google_analytics`.`analytics_date` <= 2022-12-28' and `products`.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_labels_id)) as `temp`) and `products`.`deleted_at` is null group by `google_analytics`.`analytics_date`

Input data

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversi on_rate from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date` >= 2022-12-21' and `google_analytics`.`analytics_date` <= '2022-12-28' and `products`.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_label_id` in (product_label_id)) as `t emp`) and `products`.`deleted_at` is null

Expected result

Dashboard -> Google Analytics widget -> **Activation/Deactivation parameters checkboxes**

Priority Behavior Severity Type - Not set O Medium Not set Regression Milestone **Automation Status** Is Muted Manual Actual

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled, Data interval = 6 months.
- 5. There have been sales in the last 6 months.

ared from the chart.

Click on the red "Conversion rate" check-box.

ve disappeared.

Step

Action

6. "Google Analytics" widget is added.

Steps t	Steps to reproduce		
Step	1		
Action	Pay attention to the "Google analytics" widget.		
Input da	ta		
Expecte result	1. The chart with 3 colored lines is displayed. 2. Check-boxes "Total profit", "Total revenue", "Conversion rate" are filled with check-marks. 3. Above the checkboxes there is data of the corresponding parameter for 6 months.		
Step	2		
Action	Click on the green "Total profit" check-box.		
Input da	ta		
Expecte result	 The checkbox has become empty, the green curve, which indicated the "Total profit" parameter has disappear ed from the chart. On the y-axis, the "Profit" parameter is displayed in translucent green color, the divisions along the y-axis hav e changed and correspond only to the "Revenue" parameter. 		
Step	3		
Action	Click on the blue "Total revenue" check-box.		
Input da	ta		
	1. The checkbox has become empty, the blue curve, which indicated the "Total revenue" parameter has disappe		

Expected 2. On the y-axis, the "Revenue" parameter is displayed in translucent blue color, the divisions along the y-axis ha

3. The graph displays only the red "Conversion rate" curve for 6 months.

Input data

result

- 1. The checkbox has become empty, the red curve, which indicated the "Conversion rate" parameter has disappe ared from the chart.
- Expected 2. On the right vertical axis the "Conversion rate" parameter is displayed in translucent red color, the divisions al ong the right vertical axis have disappeared.
 - 3. Graph is no longer displayed.

5 Step

Action Click on the blue "Total revenue" check-box.

Input data

1. The checkbox has become filled, the blue curve, which indicated the "Total revenue" parameter appeared on t he chart.

- Expected 2. On the y-axis, the "Revenue" parameter is displayed in blue color, the divisions along the y-axis correspond on ly to the "Revenue" parameter.
 - 3. The graph displays only the blue "Total revenue" curve for 6 months.

Dashboard -> Google Analytics widget -> Calendar work

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings Data interval = 6 months.
- 5. There have been sales in the last 6 months.
- 6. "Google Analytics" widget is added.

Step	1
Action	Pay attention to the "Google analytics" widget.
Input data	
Expected result	 The chart with 3 colored lines is displayed. Check-boxes "Total profit", "Total revenue", "Conversion rate" are filled with check-marks. Above the checkboxes there is data of the corresponding parameter for 6 months. On the x-asix dates are displayed in year-month-day format.
Step	2
Action	Click on the date "dd-mm-yyyy" next to "From".
Input data	
Expected result	Dropdown with calendar is opened. Date "From" circled in white (6 month ago).
Step	3
Action	On the "Calendar" dropdown click on the "Next" right arrow 5 times.
Input data	
Expected result	The calendar was scrolled from the 6 months ago to the last month.
Step	4
Action	Click on the "15" date.
Input data	

- 1. Date "15" circled in white.
 2. Dropdown closes.

 Expected result 4. Pay attention to the dates on the x-asix and to the "From" 5. "To" dates.

Displaying the "Google Analytics" widget when GA is disabled without DT

Check displaying the "Google Analytics" widget when GA is disabled on the Datafeeds page

Severity	Priority	Behavior	Туре
Not set	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

Log in as user/admin to the connected account.

Products have filled "Cost price" attribute.

On the Datafeeds page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled.

Dashboard page is opened.

"Google Analytics" widget is added.

Step	1
Action	On the Dashboard page pay attention to the "Google Analytics" widget
Input data	
Expected result	The "Google Analytics" widget displays relevant data
Step	2
Action	Open the Datafeeds page click on "G" Google Analitycs sign
Input data	
Expected result	"Feed's name Google Analytics" popup was displayed
Step	3
Action	Turn off "Enable Google Analytics" toogle
Input data	
Expected result	The "Enable Google Analytics" toogle is off
Step	4
Action	Click on the "Save " button
Input data	
Expected result	"Google Analytics settings saved" informational message was shown

Step	5
Action	Open the Dashboard page
Input data	
Expected result	Dashboard page is opened. Pay attention that "Google Analytics" widget is not displayed anymore

Displaying the "Google Analytics" widget when GA is disabled with DT

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

Pre-conditions

- 1. В аккаунте есть продукты.
- 2. В аккаунте включена GA на Datafeeds странице.
- 3. В аккаунте есть динамический тег "Test" с любой включенной карточкой с "Google analytics" вкладки.
- 4. На странице Dashboard добавлен виджет "Google analytics"

1		
Перейти на Datafeeds страницу и нажать на значок гугл аналитики.		
1. Открылся " <hазвание фида=""> google analytics" попап. 2. "Enable Google Analytics" бегунок активный.</hазвание>		
2		
Деактивировать "Enable Google Analytics" бегунок.		
 Открылся "Disable dynamic tags rules based on GA" попап. Отображается список динамических тегов, в которых активна хотя бы 1 карточка с гугл аналитик и Test 		
3		
Открыть Dashboard страницу.		
Виджет "Google analytics" отображается на странице.		

Dashboard -> GA widget-> "Calculate gross profit with VAT"

Displaying "Google Analytics" widget data according to the ON/OFF "Calculate gross profit with tax" settings

Severity **Priority Behavior Type** O Normal O Medium Not set Regression Milestone Automation **Status** Is Muted Manual Actual No

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled.
- 5. On the "Settings" page Option "Calculate gross profit with tax" is on.6. "Google Analytics" widget is added.

Step	1
Action	On the Dashboard page pay attention to the "Google Analytics" widget's profit data.
Input data	
Expecte result	dProfit data for the single purchase is calculated by formula including VAT % Profit= ((display price/((100+VAT)/100)) - cost price)
Step	2
Action	On Settings -> "Options" turn off the "Calculate gross profit with tax" toogle.
Input data	
Expecte result	d1. The Settings page is opened. 2. The toogle is off next to the "Calculate gross profit with tax" option. 3. "Account settings updated" and "Settings saved" information messages was shown.
Step	3
Action	Open the Dashboard page, pay attention to the "Google Analytics" profit data.
Input data	select IFNULL(SUM(google_analytics.profit * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchases), '-') as unique_sales, IFN ULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROU ND((IF(SUM(google_analytics.visitors)) * Joode analytics.visitors) = 0, 0, SUM(google_analytics.visitors) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversion_rate from `products` in ner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date` >= 2022-10-10' and `google_analytics`.`analytics_date` <= '2022-11-10' and `products`.`deleted_at` is null

Profit data for the single purchase is calculated by formula without VAT % ExpectedProfit= Display price-Cost price result Formulas are here https://docs.google.com/spreadsheets/d/1joQdP5pPlFdH7kbHmUnnBLbYub8m1TzKioHrnyOrq5Q /edit#gid=1235726146

Dashboard -> "Google Analytics" widget -> Check attributes value

Check values for "Total profit", "Total revenue", "Conversion rate", "Number of sales", "Units sold", "Total views".

Severity	Priority	Behavior	Туре
O Normal	↑ High	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

1

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled.
- 5. On the "Settings" page Option "Calculate gross profit with tax" is on.6. "Google Analytics" widget is added.

Steps to reproduce

Step	1
Action	Pay attention to the SUM "Total profit", "Total revenue", "Conversion rate", "Number of sales", "Units sold", "Total views" values on the GA widget. Check DB.
Input data	select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.vi sitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversion of the control

on_rate from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = **feed_id** and `google_analytics`.`analytics_date` >= **2022-10-10**' and `google_ analytics`.`analytics date` <= '2022-11-10' and `products`.`deleted at` is null

Expected Values are match. result

Step 2

Action

Pay attention to the "Total profit", "Total revenue", "Conversion rate", "Number of sales", "Units sold", "Total vie ws" day by day values on the GA widget graphs. Check DB.

Input data

select IFNULL(SUM(google analytics.profit without vat * google analytics.quantity), '-') as profit, IFNULL(SUM(google analytics.sales), '-') as revenue without vat, IFNULL(SUM(google analytics.unique purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchase s) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = $0,\,0,\,SUM(google_analytics.unique_purchases)\,/\,SUM(google_analytics.visitors))*100),\,2),\,{}^{\text{'}\%'}))\;as\;avg_conversi$ on_rate, google_analytics.analytics_date from `products` inner join google_analytics on `google_analytics`.`pro duct_id` = `products`.`id` where `google_analytics`.`feed_id` = **feed_id** and `google_analytics`.`analytics_dat e' >= '2022-10-10' and `google analytics`.`analytics date` <= 2022-11-10' and `products`.`deleted at is null group by `google_analytics`.`analytics_date`

Expected Values are match. result

Dashboard -> Google Analytics widget -> Check data in widget (Universal Analytics)

Проверить верность данных в которая отображается в виджете и сравнить с базой данных, написать запрос по которому быстро можно будет проверить данные

Severity	Priority	Behavior	Туре
☆ Critical	↑ High	Positive	Regression
Milestone	Automation	Status	Is Muted
March 1.0 (2023)			

1

Pre-conditions

- 1. В аккаунте включена гугл аналитика (Universal Analytics) и данные заполнены.
- 2. В аккаунт есть 5 товаров с данными гугл-аналитики, у которых:
 - brand test brand
 - Product Type test type
 - Tag test_tag
 - Dynamic Tag test__dynamic_tag
 - Label test_label
- 3. На странице "DASHBOARD" добавлен "Google Analytics" виджет.
- 4. Отрыть "DASHBOARD" страницу.

Step	1
	В "Google Analytics" виджете в календарях выбрать период 1 месяц. В "Google Analytics" виджете открыть фильтра и выбрать:
Action	 brand - test_brand Product Type - test_type Tag - test_tag Dynamic Tag - testdynamic_tag Label - test_label
Input data	
	еd. Виджет отфильтрован. 2. Период изменен.
Step	2
Action	Выполнить запрос в базе данных

UM(google_analytics.sales), '-') as total_revenue, IF(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM (google_analytics.visitors)) * 1 00), 2), '%')) as conversion_rate ,IFNULL(SUM(google_analytics.unique_purchases), '-') as numder_of_sales, IFNU LL(SUM(google_analytics.quantity), '-') as units_sold, IFNULL(SUM(google_analytics.visitors), '-') as total_views fr om `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = FEED_ID and `google_analytics`.`analytics_date` >= 'YYYYY-MM-DD' and `google_analytics`.`analytics_date` <= 'YYYYY-MM-DD' and `google_analytics`.`analytics_date` <= 'PRODUCT_TYPE_ID)) and `products_i` in (PRODUCT_TYPE_ID)) and `products_i` id` in (select * from (select `product_i d` as `id` from `products_product_labels` where `products_product_labels_ion in (PRODUCT_LABEL_ID)) as `temp`) and (JSON_SEARCH(custom_labels_data, 'one', 'PRODUCT_LABEL_ID)) as `temp`) and (IS_JSON_ARRAYS_INTERSECT(dy Trom `sphinx_products` where `sphinx_products`.`id` and (IS_JSON_ARRAYS_INTERSECT(dy

namic tag data->"\$.ids", JSON ARRAY(DYNAMIC TAG ID</spa

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as total_profit, IFNULL(S

Input data

Expected Данные соответствуют с теми, что отображаются на "DASHBOARD" странице. result

Step 3

Action В настройках аккаунта выключить опцию "Calculate gross profit with tax"

Input data

Expected result Опция выключена.

n>)))))

Step 4

Action Отрыть "DASHBOARD" страницу.

Input data

Expected Страница открыта. result

Step 5

Action Выполнить запрос в базе данных

select IFNULL(SUM(google analytics.profit * google analytics.quantity), '-') as total profit, IFNULL(SUM(google a nalytics.sales), '-') as total revenue, IF(ROUND((IF(SUM(google analytics.visitors) = 0, 0, SUM(google analytics. unique_purchases) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_ana lytics.visitors) = 0, 0, SUM(google analytics.unique purchases) / SUM(google analytics.visitors)) * 100), 2), (%)) as conversion_rate_,IFNULL(SUM(google_analytics.unique_purchases), '-') as numder_of_sales, IFNULL(SUM(google_analytics.unique_purchases), '-') le_analytics.quantity), '-') as units_sold, IFNULL(SUM(google_analytics.visitors), '-') as total_views from `products `inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`fe ed id` = FEED_ID and `google_analytics`.`product_id` is not null and `google_analytics`.`analytics_date` >= 'YYYY-M M-DD' and `google_analytics`.`analytics_date` <= 'YY YY-MM-DD' and `google_analytics`.`deleted_at` is null and `products`.`brand_id` in (BRAND_ID) and (`product_type_id` in (PRODUCT_TYPE_ID)) and `products`.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_label_id` in (PRODUCT LABEL ID)) as `temp`) and (ISON SEARCH(custom labels data, 'one', 'VALUE') IS NOT NULL) and (exists (select 1 from `sphi nx_products` where `sphinx_products`.`id` = `products`.`id` and (IS_JSON_ARRAYS_INTERSECT(dynamic_tag_d ata->"\$.ids", JSON ARRAY(DYNAMIC TAG ID)))))

Input data

 $\mathsf{Expected}_{\ensuremath{\mathsf{Д}} \mathsf{a}\mathsf{h}\mathsf{h}\mathsf{b}\mathsf{l}\mathsf{e}} \mathsf{cootbetctby} \mathsf{iot} \mathsf{c} \mathsf{temu}, \mathsf{что} \mathsf{otoбражаются} \mathsf{ha} \mathsf{"DASHBOARD"} \mathsf{ctpahule}.$

Dashboard -> Google Analytics widget -> Check data in widget (GA4)

SeverityPriorityBehaviorType★ Critical↑ HighPositiveRegressionMilestoneAutomationStatusIs MutedMarch 1.0 (2023)ManualActualNo

1

Pre-conditions

- 1. В аккаунте включена гугл аналитика (GA4) и данные заполнены.
- 2. В аккаунт есть 5 товаров с данными гугл-аналитики, у которых:
 - brand test_brand
 - Product Type test_type
 - Tag test_tag
 - Dynamic Tag test__dynamic_tag
 - Label test_label
- 3. На странице "DASHBOARD" добавлен "Google Analytics" виджет.
- 4. Опция "Calculate gross profit with tax" включена.
- 5. Отрыть "DASHBOARD" страницу.

Step	1
	В "Google Analytics" виджете в календарях выбрать период 1 месяц. В "Google Analytics" виджете открыть фильтра и выбрать:
Action	brand - test_brandProduct Type - test_type
	• Tag - test_tag
	Dynamic Tag - testdynamic_tag
	Label - test_label
Input data	
Expecte	থ্রে. Виджет отфильтрован.
	2. Период изменен.
Step	2
Action	Выполнить запрос в базе данных

Input data

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as total_profit, IFNULL(S UM(google analytics.sales), '-') as total revenue, IF(ROUND(IF(COUNT(conversion rate) = 0, 0, SUM(google anal ytics.conversion_rate) / COUNT(conversion_rate)), 2) IS NULL, '-', CONCAT(ROUND(IF(COUNT(conversion_rate) = 0, 0, SUM(google_analytics.conversion_rate) / COUNT(conversion_rate)), 2), '%')) as conversion_rate, IFNULL(SU M(google analytics.quantity), '-') as units sold, IFNULL(SUM(google analytics.visitors), '-') as total views from `p roducts` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analyti cs`.`feed_id` = FEED_ID and `google_analytics`.`product_id` is not null and `google_analytics`.`analytics_date` >= '
YYYY-MM-DD' and `google_analytics`.`analytics_date` <= ' d)">YYYY-MM-DD' and `google_analytics`.`deleted_at` is null and `products`.`brand id` in (BRAND_ID) and (`product_type_id` in (PRODUCT_TYPE_ID)) and `products`.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_label_id` in (PRODUCT_LABEL_ID)) as `temp`) and (JSON_SEARCH(custom_labels_data, 'one', 'VALUE') IS NOT NULL) and (exists (select 1 from `sphinx products` where `sphinx products`.`id` = `products`.`id` and (IS |SON ARRAYS INTERSECT(dynamic t ag data->"\$.ids", JSON ARRAY(DYNAMIC TAG ID)))))

Expected Данные соответствуют с теми, что отображаются на "DASHBOARD" странице. 3 Step Action В настройках аккаунта выключить опцию "Calculate gross profit with tax" Input data Expected Опция выключена. result Step Action Отрыть "DASHBOARD" страницу. Input data Expected Страница открыта. result Step 5

Input data

Action

Выполнить запрос в базе данных

select IFNULL(SUM(google analytics.profit * google analytics.quantity), '-') as total profit, IFNULL(SUM(google a nalytics.sales), '-') as total revenue, IF(ROUND(IF(COUNT(conversion_rate) = 0, 0, SUM(google_analytics.conver sion rate) / COUNT(conversion rate)), 2) IS NULL, '-', CONCAT(ROUND(IF(COUNT(conversion rate) = 0, 0, SUM(g oogle_analytics.conversion_rate) / COUNT(conversion_rate)), 2), '%')) as conversion_rate, IFNULL(SUM(google_a nalytics.quantity), '-') as units_sold, IFNULL(SUM(google_analytics.visitors), '-') as total_views from `products` in ner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_i $\label{eq:def} $$d = \span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and `google_analytics`.`product_id` is not be a span style = \color: var(--md-font-color-failed)">FEED_ID and \color: var(-$ ot null and `google_analytics`.`analytics_date` >= 'YYYY-MM-D D' and `google_analytics`.`analytics_date` <= 'YYYY-MM-DD' and `google_analytics`.`deleted_at` is null and `products`.`brand_id` in (BRAND ID) and (`product type id` in (PRODUCT_TYPE_ID)) and `products`.`id` in (select * from (select `product_id` as `id` from products_product_labels` where `products_product_labels`.`product_label_id` in (PRODUCT_LABEL_ID)) as `temp`) and (JSON_SEARCH(custom_labels_data, 'one', '<s pan style="color:var(--md-font-color-failed)">VALUE') IS NOT NULL) and (exists (select 1 from `sphinx_ products` where `sphinx products`.`id` = `products`.`id` and (IS JSON ARRAYS INTERSECT(dynamic tag data->"\$.ids", JSON_ARRAY(DYNAMIC_TAG_ID)))))

result

 $\mathsf{Expected}_{\mathsf{Данные}}$ данные соответствуют с теми, что отображаются на "DASHBOARD" странице.

E2E: Dashboard -> Google Analytics widget -> Filters -> "Brands"

"Google Analytics" widget displaying for selected Brand

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

1

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Account has products from minimum 3 different brands.
- 4. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 5. On the Settings page Google Analytics widgets is enabled.
- 6. Settings->Options->Calculate gross profit with tax is ON.
- 7. Dashboard page is opened.
- 8. "Google analytics " widget is added.

Steps to reproduce

Step	1
Action	On the "Google analytics" widget click on the red editing "Pen".
Input data	
Expecte result	CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Brands" search field.
Input data	
Expecte result	^d Dropdown with Brands was displayed.
Step	3
Action	Select 2 Brands, "Test1" and "Test2", by clicking on checkboxes.
Input data	
Expecte result	Selected 2 Brand's checkboxes are filled with checkmarks.
Cton	

Step 4

Action Click on the "Save" button.

Input data

ExpectedWidgets editing mode was closed. Information message with "Widget was successfully updated" was shown. result Selected brands are listed under the widget name.

Step !

Action Сравнить информацию по запросам

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversi on_rate, google_analytics.analytics_date from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date e` >= '2022-11-28' and `google_analytics`.`analytics_date` <= 2022-12-28' and `products`.`brand_id` in (b rand_id,brand_id) and `products`.`deleted_at` is null group by `google_analytics`.`analytics_date`

Input data

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchases), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors)) * 0, 0, SUM(google_analytics.unique_purchases) / SUM (google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors)) * 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversion_rate from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id_and `google_analytics`.`analytics_date` >= '2022-11-28' and `google_analytics`.`analytics_date` >= '2022-11-28' and `products`.`delet ed_at` is null

Expected result

E2E: Dashboard -> Google Analytics widget -> Filters-> "Product Types"

"Google Analytics" widget displaying for selected product types

Severity	Priority	Behavior	Туре
Not set	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Settings->Options->Calculate gross profit with tax is ON.

Account has products from minimum 2 different product types.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.

"Google Analytics" widget is added.

Step	1
Action	On the "Google Analytics" widget click on the red editing "Pen".
Input data	
Expecte result	d _{"CHOOSE} WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Product types" search field.
Input data	
Expecte result	^d Dropdown with product types was displayed.
Step	3
Action	Enter "Test1" product type's name into the search field.
Input data	
Expecte result	dL. Search fileld is filled with "Test1". 2. On the dropdown list is the "Test1"product type.

Step 4

Action Select "Test1" product type by clicking on checkbox.

Input data

Expected Selected Product type checkbox is filled with the checkmark.

Step 5

Action Click on the "Save" button.

Input data

Expected 1. Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.

On the charts there are only information and data for all products from "Test1"product type for the last 30 days

Step 6

Сравнить данные с запросами в базе.

Action

- 1. По датам и точкам.
- 2. Общие данные (сумма).

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SUM(google_analytics.unique_purchase s) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversi on_rate, google_analytics.analytics_date from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date e` >= '2022-11-28' and `google_analytics`.`analytics_date` <= 2022-12-28' and `products`.`product_type_id` in (product_type_id) and `products`.`deleted_at` is null group by `google_analytics`.`analytics_date`

Input data

select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(g
oogle_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchases), '-') as uniq
ue_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '') as views, IF(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SUM(google_analytics.unique_purchases) / SUM
(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors)) = 0, 0, SU
M(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversion_rate
from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id_and `google_analytics`.`analytics_date` >= '2022-11-28' and `google_analytics
s`.`analytics_date` <= '2022-12-28' and `products`.`product_type_id` in (product_type_id) and `products`.`
deleted_at` is null</pre>

Expected result

E2E: Dashboard -> Google Analytics widget -> Filters-> "Dynamic Tags"

"Google Analytics" widget displaying for selected Dynamic tags

Severity	Priority	Behavior	Туре
O Normal	O Medium	Not set	Other
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

1

Pre-conditions

Log in as user/admin to the connected account.
There are products with matches in the account.
Products have filled "Cost price" attribute.
Account has products with minimum 2 different dynamic tags.
On the Datafeed page Google analytics is enabled, data is connected or generated test data.
On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
"Google Analytics" widget is added.

Step	1
Action	On the "Google Analytics" widget click on the red editing "Pen".
Input data	
Expected result	"CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expected result	Dropdown with product types was displayed.
Step	3
Action	Enter "Test1" Tag's name into the search field.
Input data	
Expected result	 Search fileld is filled with "Test1". On the dropdown list is the "Test1"dynamic tag.
Step	4
Action	Select "Test1" dynamic tag by clicking on checkbox.

Input data	
Expected result	Selected dynamic tag's checkbox is filled with the checkmark.
Step	5
Action	Click on the "Save" button.
Input data	
Expected result	 Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown. On the charts there are only information and data for all products ftom "Test1" dynamic tag for the last 3 0 days.

E2E: Dashboard -> Google Analytics widget -> Filters -> "Imported tags"

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported_tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	Not set	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. Account has products with minimum 2 different ilmported Tags.
- 4. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 5. On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.
- 6. "Google Analytics" widget is added.

Step	1
Action	On the "Google Analytics" widget click on the red editing "Pen".
Input data	
Expecte result	ed _{"EDIT} GA WIDGET" filter's popup was displayed.
Step	2
Action	Click on "Tags" search field.
Input data	
Expecte result	ed Dropdown with Tags was displayed.
Step	3
Action	Click on "Imported Tags" field.
Input data	
Expecte result	ed Dropdown with Imported Tags are opened.

Step 4 Select "ImportedTag1" imported tag by clicking on checkbox. Action Input data $\begin{array}{l} \mathsf{Expected}_{\mathsf{Selected}} \, \mathsf{Imported} \, \mathsf{tag} \, \mathsf{checkbox} \, \mathsf{is} \, \mathsf{filled} \, \mathsf{with} \, \mathsf{the} \, \mathsf{checkmark}. \end{array}$ result 5 Step Action Click on the "Save" button. Input data Expected 1. Widgets editing mode was closed. Information message with "Widget was successfully updated" was shown.

On the charts there are only information and data for all products from "ImportedTag1" imported tag for the las result t 30 days. 6 Step Action Run query in DB: select IFNULL(SUM(google analytics.profit without vat * google analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchase s) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) =

Input data select IFNULL(SUM(google_analytics.profit_without_vat * google_analytics.quantity), '-') as profit, IFNULL(SUM(google_analytics.sales), '-') as revenue_without_vat, IFNULL(SUM(google_analytics.unique_purchas es), '-') as unique_sales, IFNULL(SUM(google_analytics.quantity), '-') as quantity, IFNULL(SUM(google_analytics.visitors), '-') as views, IF(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchase s) / SUM(google_analytics.visitors)) * 100), 2) IS NULL, '-', CONCAT(ROUND((IF(SUM(google_analytics.visitors) = 0, 0, SUM(google_analytics.unique_purchases) / SUM(google_analytics.visitors)) * 100), 2), '%')) as avg_conversi on_rate from `products` inner join google_analytics on `google_analytics`.`product_id` = `products`.`id` where `google_analytics`.`feed_id` = feed_id and `google_analytics`.`analytics_date` >= 'YYYYY-MM-DD' and `google_analytics`.`analytics_date` <= YYYYY-MM-DD' and `google_analytics`.`analytics_ate` >= 'span style="color:var(--md-font-color-failed)">YYYYY-MM-DD' and `google_analytics`.`deleted_at` is null and `products`.`id` in (select `product_id` as `id` from `product_imported_tags` where `product_imported_tags.`imported_tagid` in (imported_tag_id

 $\label{eq:condition} \textbf{Expected}_{\mbox{Values in the same with "Google Analytics" widget.}}$

E2E: Dashboard -> Google Analytics widget -> Filters-> "Labels"

"Google Analytics" widget displaying for selected Labels

SeverityPriorityBehaviorType- Not setO MediumNot setOtherMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

Log in as user/admin to the connected account.

There are products with matches in the account.

Products have filled "Cost price" attribute.

Account has products from minimum 2 different Labels.

On the Datafeed page Google analytics is enabled, data is connected or generated test data.

On the Settings page Google Analytics widgets is enabled, Data interval = 1 month.

"Google Analytics" widget is added.

Step	1
Action	On the "Google Analytics" widget click on the red editing "Pen".
Input data	
Expected resul	t "CHOOSE WHAT YOU WANT TO SEE" filter's popup was displayed.
Step	2
Action	Click on "Labels" search field.
Input data	
Expected resul	t Dropdown with Labels was displayed.
Step	3
Action	Select "Test1" Label by clicking on checkbox.
Input data	
Expected resul	tSelected Label's checkbox is filled with the checkmark
Step	4
Action	Click on the "Save" button.
Input data	

Widgets editing mode was closed.
 Information message with "Widget was successfully updated" was shown.
 On the charts there are only information and data for all products from "Test1"Label for the last 30 days

E2E: Dashboard -> Google Analytics widget -> Filters-> Reset to default

Reset to default custom filtering on the "Google Analytics" widget

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted

3

Pre-conditions

- 1. There are products with matches in the account.
- 2. Products have filled "Cost price" attribute.
- 3. On the Datafeed page Google analytics is enabled, data is connected or generated test data.
- 4. On the Settings page Google Analytics widgets is enabled.
- 5. Dashboard page is opened.
- 6. ""Google Analytics"" widget is added and filtered by custom parameters (by Brands, Tags, Dynamic tags, Product types, Labels).

Step	1
Action	Run a query in the database
Input data	SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
Expected result	In the fields params displayed JSON which displayed your filters: The result in the database is the same as on the page { "origin": null, "brandsids": [brand_id_1], "labelsids": [label_id_1], "dynamicTags": [dynamic_tag_id_1], "customLabels": ["name"], "productTypesIds": [product_type_id_1] }
Step	2
Action	Click on the red pen "Edit" button in the upper right corner on the ""Google Analytics"" widget.
Input data	
Expected result	"Choose what you want to see" pop-up is displayed.

Step	3
Action	Click on "Reset to default" button.
Input data	
Expected result	"Widget was successfully updated" information message was displayed.
Step	4
Action	Click on the red cross in the top right corner.
Input data	
Expected result	"Google Analytics" widget is displaying in the defoult state, for all products, data interval matches the setting.
Step	5
Action	Run a query in the database
Input data	SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
	In the fields params displayed JSON which displayed your filters: The result in the database is the same as on the page {
Expected result	"origin": null, "brandslds": [], "labelslds": [], "dynamicTags": [], "customLabels": [], "productTypesIds": [] }

Dashboard -> Price changes widget -> Activation/Deactivation parameters checkboxes (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumPositiveRegressionMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

result

- 1. В аккаунте есть продукты с матчами.
- 2. У некоторых конкурентов и товаров ранее изменялись цены.
- 3. На странице "Dashboard" добавлен "Price changes" виджет.
- 4. Открыть страницу "Dashboard".

Steps to reproduce

Step	1
Action	Обратить внимание на «Price changes» виджет
Input data	
Expecte result	dl. Отображается диаграмма с 3 цветными линиями. 2. Чекбоксы "Total price changes", "Price changes up", "Price changes down" заполнены галочками.
Step	2
Action	Нажать на "Total price changes" чекбокс.
Input data	
Expecte result	 Чекбокс стал пустым, с графика исчезла зеленая кривая, которая указывала на параметр "Total price changes". По оси Y параметр "Total price changes" отображается полупрозрачным зеленым цветом, деления по оси Y изменились и соответствуют только параметрам "Price changes up" и "Price changes down". В тултипе при наведении на график не отображается параметр Total price changes". В тултипе при наведении отображаются только "Price changes up", "Price changes down", "Own price changes", "Own price changes up", "Own price changes" параметры.
Step	3
Action	Нажать на "Price changes up" чекбокс.
Input data	
	1. Чекбокс стал пустым, с графика исчезла синяя кривая, которая указывала на параметр "Price changes up". 2. По оси Y параметр "Price changes up" отображается полупрозрачным зеленым цветом, деления по оси Y

изменились и соответствуют только параметру "Price changes down".

nges up", "Own price changes" параметры.

3. В тултипе при наведении на график не отображается параметр "Price changes up".

4. В тултипе при наведении отображаются только "Price changes down", "Own price changes", "Own price cha

Ste	р	4

Action Нажать на "Price changes down" чекбокс.

Input data

1. Чекбокс стал пустым, с графика исчезла красная кривая, которая указывала на параметр "Price changes down".

result

- Expected². По оси Y параметр "Price changes down" отображается полупрозрачным зеленым цветом, деления по оси Y изменились и соответствуют только параметру "Price changes down".
 - 3. В тултипе при наведении на график не отображается параметр "Price changes down".
 - 4. В тултипе при наведении отображаются только "Own price changes", "Own price changes up", "Own price c hanges" параметры.

Dashboard -> Price changes widget -> Check data in widget (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumPositiveRegressionMilestoneAutomationStatusIs Muted-ManualActualNo

1

Pre-conditions

- 1. В аккаунте есть 5 товаров и матчей к ним, у которых есть данные о изменении цен, у них есть атрибуты:
 - brand test_brand
 - Product Type test_type
 - Tag test_tag
 - Dynamic Tag test__dynamic_tag
 - Label test label
- 2. На странице "Dashboard" добавлен "Price changes" виджет.
- 3. Открыть страницу "Dashboard".

Steps to reproduce

Step	1
	В "Price changes" виджете в календарях выбрать период 1 месяц. В "Price changes" виджете открыть фильтра и выбрать:
Action	 brand - test_brand Product Type - test_type Tag - test_tag Dynamic Tag - test_dynamic_tag Label - test_label
Input data	

Step 2

Action Выполнить запрос в базе данных

select

JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn.

product price history.history date

Input data from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `products` where `products`.`brand_id` in (PRAND_ID) and (`product_type_id` in (PRODUCT_TYPE_ID)) and `product_s`.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_label_id` in (PRODUCT_LABEL_ID)) as `temp`) and (JSON_SEARCH(custom_labels_data, 'one', 'VALUE') IS NOT NULL) and (exists (select 1 from `sphinx_products` where `sphinx_products`.`id` = `products`.`id` and (IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->"\$.ids", JSON_ARRAY(DYNAMIC_TAG_ID))))) and `products`.` deleted_at` is null) and `product_price_history`. history_date` >= 'YYYY-MM-DD' and `product_price_history`. history_date` <= 'YYYY-MM-DD' and `product_price_history`. field_name` = 'display_price' and product_price_history. history_date` <= 'color:var(--md-font-color-failed)">Product_price_history. history_date` <= 'display_price' and product_price_history. history_date` <= 'color:var(--md-font-color-failed)">Product_price_history. history_date` <= 'color:var(--md-font-color

Expected. Количество повышений цены, понижений цены и общее количество для сообственных продуктов совпад result ают с виджетом.

Step 3

Action Выполнить запрос в базе данных

select

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competitor_changes up,

 $IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), \ 0) \ as \ competitor_changes_down$

Input data , `competitor_price_changes_product` .`history_date` from `competitor_price_changes_product` where `compet itor_price_changes_product` .`product_id` in (select `products` .`id` from `products` where `products` .`brand_id `in (BRAND_ID) and (`product_type_id` in (PRODUCT_TYPE_ID)) and `products` .`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels` .`product_label_id` in (PRODUCT_LABEL_ID)) as `temp`) and (JSON_SEARCH(custom_labels_data, 'one', 'VALUE') IS NOT NULL) and (exists (select 1 from `sphinx_products` where `sphinx_products` .`id` = `products` .`id` and (IS_JSON_ARRAYS_I NTERSECT(dynamic_tag_data->"\$.ids", JSON_ARRAY(DYNAMIC_TAG_ID))))) and `products` .` deleted_at` is null) and `competitor_price_changes_product` .`history_date` >= 'YYYYY-MM-DD' and `competitor_price_changes_product` .`history_date` <= 'YYYYY-MM-DD' and competitor_price_changes_product` .`history_date` <= 'YYYYY-MM-DD

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result eтом.

Dashboard -> Price changes -> Calendar work

новая задача 4915

SeverityPriorityBehaviorTypeO NormalO MediumNot setRegressionMilestoneAutomationStatusIs Muted-ManualActualNo

1

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У матчей есть история изменения цен в период > месяца

Нажать на дату "dd-mm-уууу" возле "to".

2

- 3. "Price changes" виджет добавлен на Dashboard.
- 4. Включена Гугл аналитика в аккаунте.
- 5. Открыть Dashboard.

Steps to reproduce

Action

Step	1
Action	Обратить внимание на дефолтный период на "Price changes" виджете.
Input data	
Expected result	 В "From" отображается дата, что ровна недели назад. В "То" отображается вчерашняя дата. Точки на графике отображают даты выбранного периода.
Step	2
Action	Нажать на дату "dd-mm-уууу" возле "From".
Input data	
Expected result	1. Календарь открылся. 2. Дата "From" отображается в белом кружке, дата ровна дню - неделя назад.
Step	3
Action	Выбрать в календаре любую дату в прошлом.
Input data	
Expected result	 Календарь закрылся. График изменился, отображаются точки с выбранной даты. В "From" отображается выбранная дата. В "То" отображается вчерашняя дата. В даты, где не собрана информация по изменениям цен конкурентов - отображается "0 Price chan ges".
Step	4

Input data Expected 1. Календарь открылся. result 2. Дата "to" отображается в белом кружке, дата ровна- вчерашнему дню. 5 Step Action Выбрать в календаре дату в прошлом, что не раньше "From". Input data 1. Календарь закрылся. 2. График изменился, отображаются точки в диапазоне выбранных дат. Expected 3. В "to" отображается выбранная дата. result 4. В даты, где не собрана информация по изменениям цен конкурентов - отображается "0 Price chan ges". 6 Step 1. Перезагрузить станицу. Action 2. Обратить внимание на период на "Price changes" виджете. Input data 1. Страница перезагрузилась. Expected 2. В "From" отображается дата, что ровна недели назад. 3. В "То" отображается вчерашняя дата. result 4. Точки на графике отображают даты выбранного периода. 7 Step 1. В Settings изменить период для Google analytics. Action 2. Проверить, что выставленный период не повлиял на диапазон "Price changes" виджета.

Период для Google analytics изменен и не повлиял на диапазон времени на "Price changes" виджете.

6 months

Input data

Expected

result

E2E: Dashboard -> Price changes widget -> Filters -> "Brands" (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumNot setFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 бренда.
- 3. На странице "Dashboard" добавлен "Price changes" виджет.
- 4. Открыть страницу "Dashboard".

Steps to reproduce

Step

Action

Выполнить запрос БД

Step	1
Action	 Перейти в окно редактирования виджета и в дропдауне брендов выбрать Brand_1. Сохранить изменения.
Input data	
Expecte result	ed 1. Виджет отфильтрован по <i>Brand_1 параметру.</i>
Step	2
Action	Выполнить запрос БД
Input data	select JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn, JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) < IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_up , product_price_history where `product_price_history`.`product_id` in (select `products`.`id` from `products` where `products`.`brand_id` in (BRAND_ID) and `products`.`brand_id` in (BRAND_ID) and `products`.`brand_id` in (<pan style="color:var(md-font-color-failed)">YYYY-MM-DD' and `product_price_history`.`history_date` <= 'YYYY-MM-DD' and `product_price_history`.`field_name` = 'display_price' and pr oduct_price_history.account_id = ACCOUNT_ID group b y `product_price_history`.`history_date`</pan>

Input data

select IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competit or_changes_up, IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), 0) as competitor_changes_down , `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`product_id` in (select `products`.`id` from `product s` where `products`.`brand_id` in (BRAND_ID) and `products`.`deleted_at` is null) and `competitor_price_changes_product`.`history_date` >= 'YYYYY-MM-DD' and `competitor_price_changes_product`.`history_date` <= 'YYYYY-MM-DD' and competitor_price_changes_product. account_id = ACCOUNT_ID group by `competitor_price_changes_product`.`history_date`

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result eтом.

E2E: Dashboard -> Price changes widget -> Filters -> "Product types" (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumNot setFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

3

result

Step

Action

ают с виджетом.

Выполнить запрос БД

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 продукт типа.
- 3. На странице "Dashboard" добавлен "Price changes" виджет.
- 4. Открыть страницу "Dashboard".

Steps to reproduce

Step	1
Action	 Перейти в окно редактирования виджета и в дропдауне продукт типов выбрать Product_type_1. Сохранить изменения.
Input data	
Expecte result	ed 1. Виджет отфильтрован по <i>Product_type_1 параметру.</i>
Step	2
Action	Выполнить запрос БД
Input data	JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn, JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) < IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_up , product_price_history history_date from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `products` where `products`.`product_type_id` in (PRODUCT_TYPE_ID) and `products`.`deleted_at` is null) and `product_price_history`.`history_date` >= ' YYYYY-MM-DD ' and `product_price_history`.`history_date` <= ' YYYY-MM-DD ' and `product_price_history`.`field_name` = 'disp lay_price' and product_price_history.account_id = ACCOUNT_I D group by `product_price_history`.`history_date`

Expected. Количество повышений цены, понижений цены и общее количество для сообственных продуктов совпад

Input data

select IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competit or_changes_up, IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), 0) as competitor_changes_down, `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`product_id` in (select `products`.`id` from `product s` where `products`.`product_type_id` in (PRODUCT_TYPE_ID) and `products`.`deleted_at` is null) and `competitor_price_changes_product`.`history_date` >= 'YYYYY-MM-DD' and `competitor_price_changes_product`.`history_date` <= 'YYYYY-MM-DD' and competitor_price_changes_product.account_id = ACCOUNT_ID group by `competitor_price_changes_product`.`history_date`

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result eтом.

E2E: Dashboard -> Price changes widget -> Filters -> "Tags" (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumNot setRegressionMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 тега.
- 3. У своих товаров и конкурентов минимум 1 день назад изменялись цены.
- 4. На странице "Dashboard" добавлен "Price changes" виджет.
- 5. Открыть страницу "Dashboard".

Steps to reproduce

Step	1
Action	 Перейти в окно редактирования виджета и в дропдауне тегов выбрать <i>Tag_1</i>. Сохранить изменения.
Input data	
Expecte result	ed 1. Виджет отфильтрован по <i>Tag_1 параметру.</i>
Step	2

Action Выполнить запрос БД

select

JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn,

Input data $product_price_history.history_date$

from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `product ts` where `products'.`id` in (select * from (select `product_id` as `id` from `products_product_labels` where `products_product_labels`.`product_label_id` in (TAG_ID)) as `temp`) and `products`.` deleted_at` is null) and `product_price_history`.`history_date` >= 'YYYYY-MM-DD' and `product_price_history`.`history_date` <= 'YYYYY-MM-DD' and `product_price_history`.` field_name` = 'disp lay_price' and product_price_history.account_id = ACCOUNT_I D group by `product_price_history..` history_date`

Expected. Количество повышений цены, понижений цены и общее количество для сообственных продуктов совпад result ают с виджетом.

Step 3

Action Выполнить запрос БД

select

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competitor_changes up,

 $IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), \ 0) \ as \ competitor_changes \ down$

Input data ', 'competitor_price_changes_product'.' history_date' from 'competitor_price_changes_product' where 'compet itor_price_changes_product'.' product_id' in (select 'products'.' id' from 'products' where 'products'.' id' in (select * from (select 'product_id' as 'id' from 'products_product_labels' where 'products_product_labels'.' product_label_id' in (TAG_ID)) as 'temp') and 'products'. 'deleted_at' is null) and 'competitor_price_changes_product'.' history_date' >= 'YYYY-MM-DD' and 'competitor_price_changes_product'.' history_date' <= 'YYYY-MM-DD' and competitor_price_changes_product.account_id = ACCOUNT_ID group by 'competitor_price_changes_product'.' history_date'

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result етом.

E2E: Dashboard -> Price changes widget -> Filters -> "Dynamic tags" (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumNot setFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

3

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 динамических тега.
- 3. На странице "Dashboard" добавлен "Price changes" виджет.
- 4. Открыть страницу "Dashboard".

ают с виджетом.

Выполнить запрос БД

result

Step

Action

3

Steps to reproduce

Step	1
Action	 Перейти в окно редактирования виджета и в дропдауне динамических тегов выбрать <i>DT_1</i>. Сохранить изменения.
nput data	
Expecte result	ed 1. Виджет отфильтрован по <i>DT_1 параметру.</i>
Step	2
Action	Выполнить запрос БД
	select JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn, JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) < IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_up
nput data	product_price_history.history_date from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `products` where (exists (select 1 from `sphinx_products` where `sphinx_products`.`id` = `products`.`id` and (IS_JSON_ARRAYS_INTERSECT(dynamic_tag_data->"\$.ids", JSON_ARRAY(DT_ID))))) and `products`.`deleted_at` is null) and `product_price_history`.`history_date` >= ' YYYYY-MM-DD ' and `product_price_history`.`history_date` <= '< span style="color:var(md-font-color-failed)">YYYYY-MM-DD' and `product_price_history`.`field_name` = 'display_price' and product_price_history.account_id = ACCO UNT_ID group by `product_price_history`.`history_date`

select

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competitor_changes up,

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), 0) as competitor_c hanges down

Input data ", 'competitor_price_changes_product'.' history_date' from 'competitor_price_changes_product' where 'compet itor_price_changes_product'.' product_id' in (select 'products'.' id' from 'products' where (exists (select 1 from 'sphinx_products' where 'sphinx_products'.' id' = 'products'.' id' and (IS_JSON_ARRAYS_INTERSECT(dynamic_t ag_data->"\$.ids", JSON_ARRAY(DT_ID))))) and 'products'.' deleted_at' is null) and 'competitor_price_changes_product'.' history_date' >= 'YYYY-MM-DD' and 'competitor_price_changes_product'.' history_date' <= 'YYYY-MM-DD' and competitor_price_changes_product.acco unt_id = ACCOUNT_ID group by 'competitor_price_changes_product'.' history_date'

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result етом.

E2E: Dashboard -> Price changes widget -> Filters -> "Imported tags"

- Для того, чтобы у продукта было более 1 импорт-тега теги должны быть представлены в виде атрибута массива ОБЯЗАТЕЛЬНО с именем "imported tags" в любом XML файле.
- Для Шопифай фидов сразу же импортируется массив "imported tags".
- Для CSV пока что доступен только 1 импортный тег для одного продукта. (любая колонка может быть помещена в "imported_tags" ключ в маппинге)

Важно: импорт-теги нацелены на текущий момент только на Шопифай.

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Functional
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No
1	1	1	3

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 импорт-тега.
- 3. У своих товаров и конкурентов минимум 1 день назад изменялись цены.
- 4. На странице "Dashboard" добавлен "Price changes" виджет.
- 5. Открыть страницу "Dashboard".

Steps to reproduce

Step	1				
Action	1. Перейти в окно редактирования виджета и в дропдауне тегов выбрать <i>Imported_Tag_1.</i> 2. Сохранить изменения.				
Input data					
Expecte result	ed 1. Виджет отфильтрован по <i>Imported_Tag_1. параметру.</i>				
Step	2				
Action	Выполнить запрос БД				
	select JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn, JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) < IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_up				
Input data	product_price_history.history_date from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `products` where `products`.`id` in (select * from (select `product_id` as `id` from `product_imported_tags` where `product_imported_tags`.`imported_tag_id` in (imported_tag_id				

Expected. Количество повышений цены, понижений цены и общее количество для сообственных продуктов совпад result ают с виджетом.

nt_id group by `product_price_history`.`history_date`;

pan>)) as `temp`) and `products`. deleted_at` is null) and `product_price_history`. history_date` >= 'YYYY-MM-DD' and `product_price_history`.`history_date` <= '< span style="color:var(--md-font-color-failed)">YYYY-MM-DD' and `product_price_history`.`field_name` = 'display price' and product price history.account id = account id = <span style="col

Step

3

Action Выполнить запрос БД

Input data

select IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competit or_changes_up, IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), 0) as competitor_changes_down, `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product`.`product_id` in (select `products`.`id` from `product s` where `products`.`id` in (select * from (select `product_id` as `id` from `product_imported_tags` where `product_imported_tags`.`imported_tag_id` in (imported_tag_id)) as `temp`) and `products`.`deleted_at` is null) and `competitor_price_changes_product`.`history_date` >= 'YYYY-MM-DD' and `competitor_price_changes_product`.`history_date` <= 'YYYY-MM-DD' and competitor_price_changes_product.account_id = account_id group by `competitor_price_changes_product`.`history_date`;

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result етом.

E2E: Dashboard -> Price changes widget -> Filters -> "Labels" (Admin/RA)

SeverityPriorityBehaviorTypeO NormalO MediumNot setFunctionalMilestoneAutomationStatusIs Muted-ManualActualNo

3

Step

Action

3

Выполнить запрос БД

Pre-conditions

- 1. В аккаунт есть товары с матчами.
- 2. У товаров есть как минимум 3 кастомных лейбла.
- 3. На странице "Dashboard" добавлен "Price changes" виджет.
- 4. Открыть страницу "Dashboard".

Steps to reproduce

Step	1
Action	 Перейти в окно редактирования виджета и в дропдауне кастомных лейблов выбрать Label_1. Сохранить изменения.
Input data	
Expecte result	ed 1. Виджет отфильтрован по <i>Label_1 параметру.</i>
Step	2
Action	Выполнить запрос БД
	select JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) > IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_do wn, JSON_LENGTH(CONCAT('[', GROUP_CONCAT(DISTINCT(IF(IFNULL(product_price_history.clear_old_price, 0) < IFN ULL(product_price_history.clear_new_price, 0), product_price_history.product_id, null))), ']')) as own_changes_up
Input data	product_price_history.history_date from `product_price_history` where `product_price_history`.`product_id` in (select `products`.`id` from `products` where (JSON_SEARCH(custom_labels_data, 'one', ' LABEL_NA ME ') IS NOT NULL) and `products`.`deleted_at` is null) and `product_price_history`.`history_date` >= ' YYYY-MM-DD ' and `product_price_history`.`history_dat e` <= ' YYYY-MM-DD ' and `product_price_history`.`fiel d_name` = 'display_price' and product_price_history.account_id = ACCOUNT_ID group by `product_price_history`.`history_date`
Expecte result	ed. Количество повышений цены, понижений цены и общее количество для сообственных продуктов совпад ают с виджетом.

select

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_up_changes)), 0) as competitor_changes up.

IFNULL(SUM(JSON_LENGTH(competitor_price_changes_product.competitors_down_changes)), 0) as competitor_c hanges down

Input data , `competitor_price_changes_product`.`history_date` from `competitor_price_changes_product` where `compet itor_price_changes_product`.`product_id` in (select `products`.`id` from `products` where (JSON_SEARCH(custo m_labels_data, 'one', 'LABEL_NAME') IS NOT NULL) and `products`.` deleted_at` is null) and `competitor_price_changes_product`.`history_date` >= 'YYYYY-MM-DD' and `competitor_price_changes_product`.`history_date` <= 'YYYYY-MM-DD' and competitor_price_changes_product.account_id = ACCOUNT_ID group by `competitor_price_changes_product`.`history_date` <= 'changes_product`.`history_date`

Expected. Количество повышений цены, понижений цены и общее количество для конкурентов совпадают с видж result eтом.

E2E: Dashboard -> Price changes widget -> Reset to default (Admin/RA)

Severity	Priority	Behavior	Туре
O Normal	O Medium	Positive	Regression
Milestone	Automation	Status	Is Muted
-	Manual	Actual	No

3

Pre-conditions

- 1. В аккаунте есть продукты с матчами.
- 2. У некоторых конкурентов и товаров ранее изменялись цены.
- 3. На странице "Dashboard" добавлен "Price changes" виджет отфильтрован по Brands, Tags, Dynamic tags, Product types, Labels.

Steps to reproduce

```
1
Step
Action
                    Выполнить запрос в БД
                    SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
Input data
                   В параметрах отображаются выбранные фильтра.
                    "origin": null,
                    "brandsIds": [
                    brand_id_1
                    "labelsIds": [
                    label_id_1
Expected result ], "dynamicTags": [
                    dynamic_tag_id_1
                    "customLabels": [
                    "name"
                    "productTypesIds": [
                    product_type_id_1
Step
                   2
Action
                    Открыть режим редактирования виджета и нажать на "Reset to default" кнопку.
Input data
Expected result ^{1.} Отображается "Widget was successfully updated" сообщение. ^{2.} В виджете сбросились фильтра, отображаются значения для всех продуктов.
                   3
Step
Action
                   Выполнить запрос в БД
Input data
                    SELECT * FROM `widgets` WHERE `account_id` = account_id and user_id = user_id AND id = id_your_widget
```

```
В параметрах виджета отображается:

{
    "origin": null,
    "brandslds": [],
    "labelslds": [],
    "dynamicTags": [],
    "customLabels": [],
    "productTypeslds": []
}
```