### プロシージャ

#### 概要

ログプレッソ・ソナーにおけるプロシージャは、従来のDBMS環境におけるプロシージャと同様の役割を果たします。複雑なクエリをモジュール化し、異なるパラメータ値を渡すことで同じ操作を簡単に実行できます。例えば、特定のテーブルで特定のIPアドレスを検索する複雑なクエリを頻繁に実行する場合、IPアドレスを入力パラメータとして受け取るプロシージャを定義しておけば、毎回長いクエリを記述せずに迅速かつ効率的に同じタスクを実行できます。

プロシージャの主な利点は以下の通りです。

* **再利用性と保守性の向上**: 特定の機能を提供するクエリをプロシージャとしてモジュール化することで、再利用性が高まります。利用者はプロシージャ名とパラメータのみを把握すればよく、繰り返しクエリを記述する必要がなくなり、クエリ文の保守も容易になります。
* **セキュリティの強化**: [dbquery](https://docs.logpresso.comnull)、[ftp](https://docs.logpresso.comnull)、[sftp](https://docs.logpresso.comnull)など外部システムへアクセスするコマンドは、[接続プロファイル](https://docs.logpresso.comnull)の権限が必要です。ユーザーに接続プロファイルへの直接アクセスを許可すると、外部システム上で任意の操作が可能となり危険です。管理者権限が必要なコマンドや、ローカル・リモートホスト上で無制限の操作が可能なコマンドをプロシージャにカプセル化することで、システム全体の管理者権限を付与せずにユーザー権限を管理できます。この方法は、元データの一部へのアクセス制限や機微情報のマスキングにも応用できます。
* **ログプレッソ・ソナー構成情報へのアクセス**: ログプレッソ・ソナーのシステムテーブルへのアクセスには管理者権限が必要です。ユーザーがシステム構成情報へアクセスする必要がある場合、プロシージャを通じて限定的なアクセスを提供することで、全権限を付与せずに運用できます。

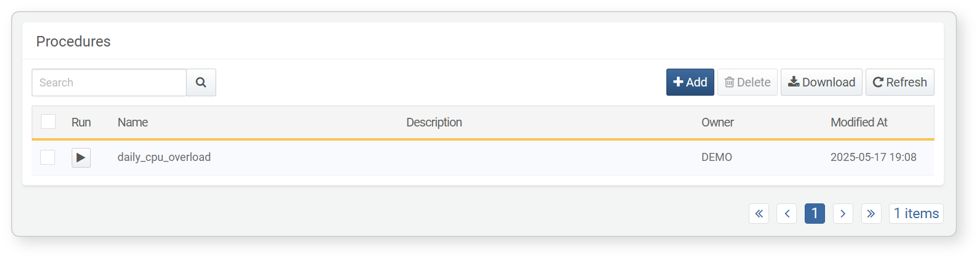
注意事項

パラメータ利用時は以下の点にご注意ください。

* パラメータは、フィルタ式などクエリ内の特定の箇所でのみ使用できます。
* クエリコマンド名やテーブル名にはパラメータを使用できません。
* これはクエリ実行の安定性確保およびSQLインジェクション攻撃防止のための制約です。

#### プロシージャ一覧の表示

**分析 > プロシージャ** でプロシージャの一覧を表示・検索できます。



* **実行**: プロシージャを実行
* **名前**: プロシージャ名
* **説明**: プロシージャの説明
* **所有者**: プロシージャを作成したユーザー
* **最終変更日**: プロシージャの最終変更（または作成）日時

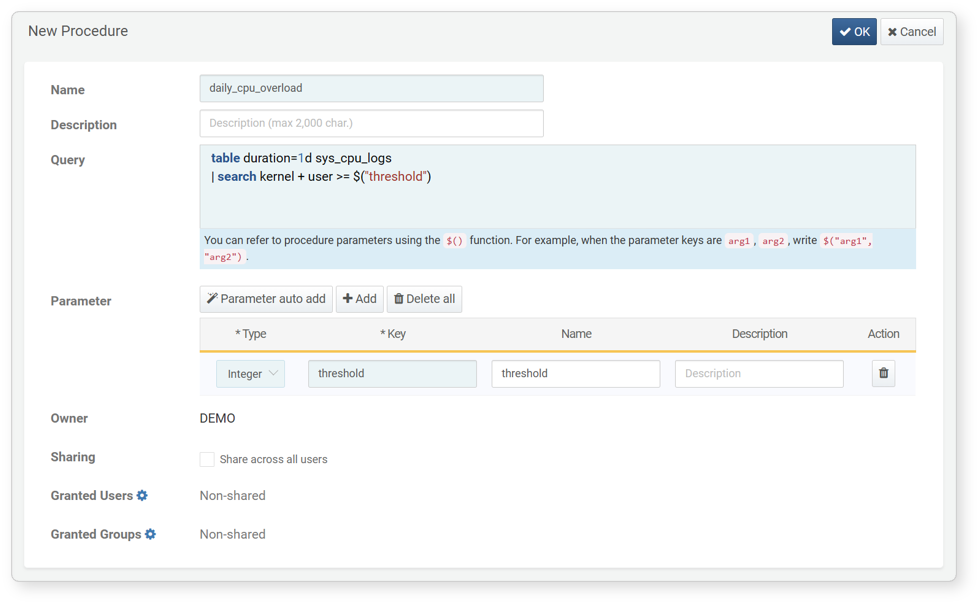
特定のプロシージャを探す場合は、ツールバーの検索ツールを利用してください。入力したキーワードが**名前**または**説明**に含まれるプロシージャを検索します。検索は大文字・小文字を区別しません。

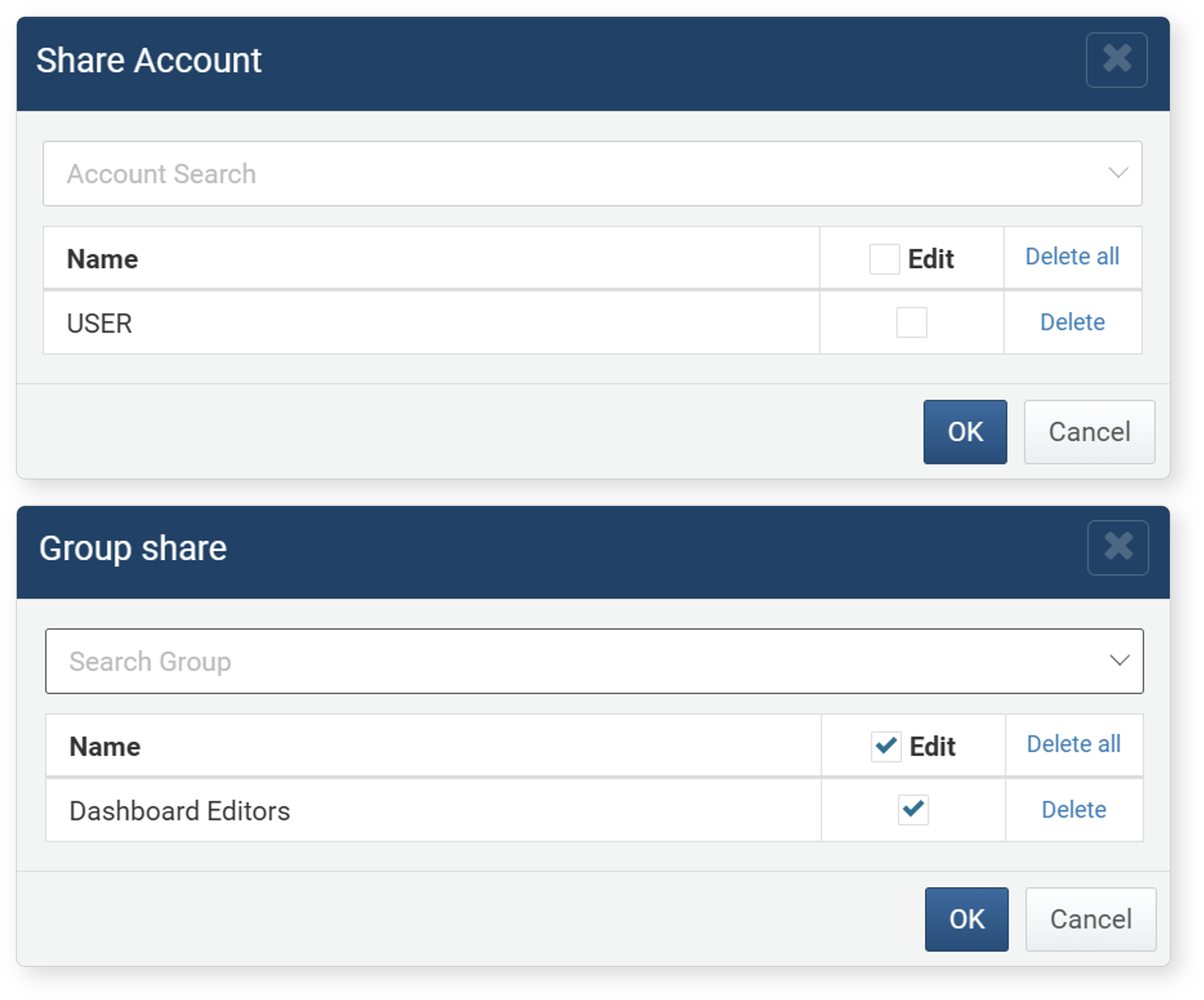
#### プロシージャの作成

プロシージャを作成するには、以下の手順を実行します。

プロシージャ一覧のツールバーで**追加**をクリックします。

**プロシージャ追加**画面で必要事項を入力し、**OK**をクリックします。



* **名前**: プロシージャ名。英数字およびアンダースコア（\_）のみ使用可能です。
* **説明**: プロシージャの詳細説明（最大2,000文字）。
* **クエリ**: プロシージャで実行するクエリを入力します。
* **パラメータ**: クエリパラメータ。
* **パラメータ自動追加**: クエリ内にパラメータ（例：$("arg1")）が含まれている場合、このボタンをクリックすると自動検出してリストに追加します。
* **追加**: クエリパラメータを手動で追加します。新しい行が表示されます。
* **型**: パラメータのデータ型
* **キー**: パラメータキー。クエリ内で使用する名称と一致させてください。
* **表示名**: パラメータの表示名
* **説明**: クエリパラメータの説明
* **操作**:  アイコンをクリックすると該当パラメータ行を削除します。
* **全削除**: すべてのパラメータ項目を削除します。
* **共有**: すべてのユーザーに閲覧・実行権限を付与します。
* **許可ユーザー**/**許可グループ**: プロシージャの権限を付与するユーザーまたはユーザーグループのリスト。設定アイコンをクリックすると**許可ユーザー**/**許可グループ**ダイアログが開き、ユーザーやグループを選択できます。**編集**チェックボックスをオンにすると、修正・削除権限も付与されます。
* 

プロシージャが他のオブジェクト（例：スケジュールクエリ、ウィジェット）から参照されている場合、保守性向上のためプロシージャの説明欄にその旨を記載することを推奨します。アクセス権限はユーザーおよびグループの共有設定に基づきます。クラスター管理者および管理者は、共有設定に関わらずプロシージャの追加・修正・削除が可能です。

プロシージャクエリ

プロシージャで使用するクエリは、実行時にユーザーから渡されるクエリパラメータを[$()](https://docs.logpresso.comnull)参照関数で参照できます。例：

# このクエリを使ったプロシージャ名を cpu\_overload とします。| table duration=1d sys\_cpu\_logs| search kernel + user >= $("threshold")

* threshold は、kernel と user フィールドの合計値と比較するための入力パラメータです。

プロシージャ作成時によくある誤りとして、$()関数がテキスト置換を行うマクロのように動作すると誤解されることがあります。実際には、$()はクエリコマンド内で式が許可されている箇所でのみ使用できます。例えば、以下のようなクエリは無効です。なぜなら[dbquery](https://docs.logpresso.comnull)は任意のSQL文を式として受け付けないためです。

dbquery USERDB $("sql")

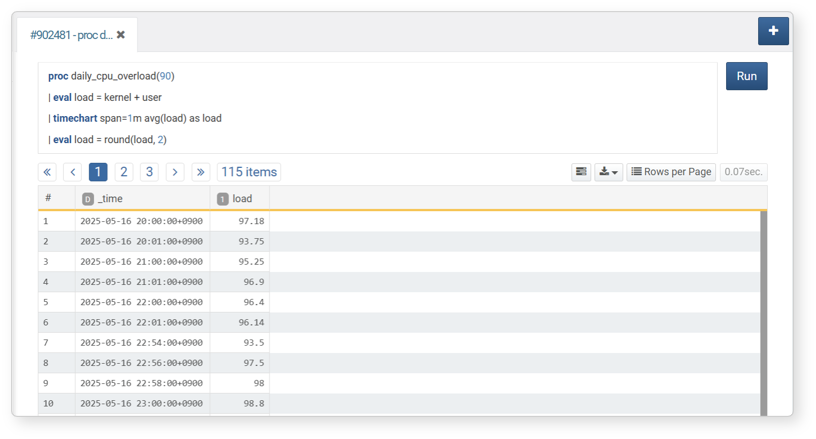
#### プロシージャの実行

プロシージャを実行する方法は2通りあります。

* クエリ内で[proc](https://docs.logpresso.comnull)コマンドを使って呼び出す
* プロシージャ一覧から**実行**をクリックする

クエリによるプロシージャ呼び出し

プロシージャは、クエリ入力が可能なほとんどの画面で実行できます。以下は**分析 > クエリ**からプロシージャを実行する例です。



cpu\_overloadプロシージャを呼び出すには：

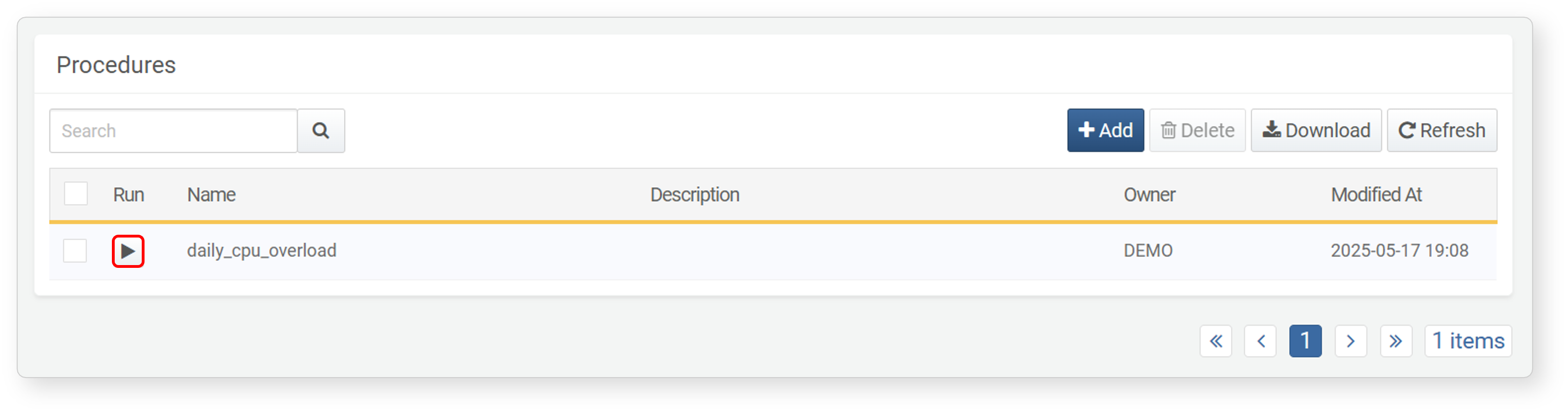
proc cpu\_overload(90)

* 括弧内の値は、プロシージャで定義されたクエリパラメータthresholdに渡されます。
* クエリパラメータがない場合は、空の括弧で呼び出します：proc cpu\_overload()。

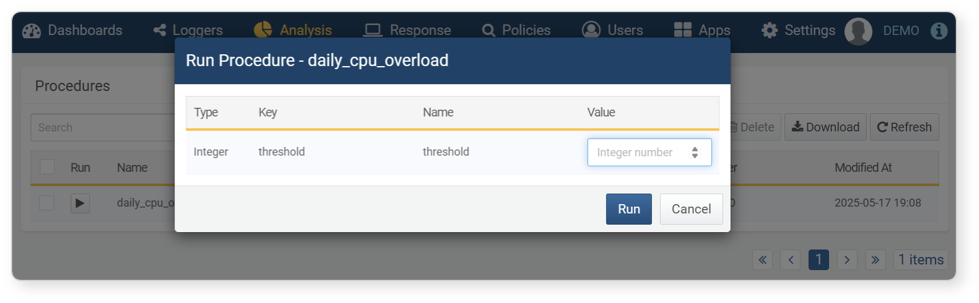
プロシージャの手動実行

プロシージャ一覧から実行するには：

[プロシージャ一覧](https://docs.logpresso.comnull)で実行したいプロシージャの**▶**アイコンをクリックします。



**プロシージャ実行**ダイアログで必要なパラメータ値を入力し、**実行**をクリックします。入力値が**型**に合致していることを確認してください。



**分析 > クエリ**画面に遷移し、実行結果が表示されます。



#### プロシージャの編集

プロシージャを編集するには：

[プロシージャ一覧](https://docs.logpresso.comnull)で編集したいプロシージャの**名前**をクリックします。

**プロシージャ編集**画面で該当項目を修正し、**OK**をクリックします。

編集可能な項目の詳細は[プロシージャ追加](https://docs.logpresso.comnull)を参照してください。

#### プロシージャの削除

プロシージャを削除するには：

[プロシージャ一覧](https://docs.logpresso.comnull)で削除したいプロシージャのチェックボックスを選択します。

ツールバーの**削除**をクリックします。

**プロシージャ削除**ダイアログで削除対象のプロシージャ一覧を確認し、**削除**をクリックします。操作を中止する場合は**キャンセル**をクリックしてください。

プロシージャを削除する前に、他の機能（例：ウィジェット、スケジュールクエリ）で使用されていないか必ず確認してください。プロシージャを削除すると、該当プロシージャを呼び出している他の機能が正常に動作しなくなる場合があります。