Coding guidelines

- JavaScript: do NOT translate text!
  Instead of
  $.mage.__('Attention')

  just use
  `Attention`

  Frans is an English-only website for now.

- Don't use `INNER JOIN`
  Use `JOIN`. Keep it simple. `INNER JOIN` is the same as `JOIN`
  Don't use `LEFT OUTER JOIN`
  Just use `LEFT JOIN`. Keep it simple. `LEFT OUTER JOIN` is the same as `LEFT JOIN`
- Table design, ID column:
  Every table must have int identity unsigned pk column, named `id`

  ```
  <table name="frans_stripe_reauth_log" resource="default" engine="innodb" comment="Stripe Reauthorization Logs">
      <column xsi:type="int" name="id" padding="10" unsigned="true" nullable="false" identity="true" comment="ID"/>
  ```

  Keep column name simple. Instead of log_id, order_item_id, just name the column `id`. The table name already tells if it's a log or an order_item. No need to repeat that in the column name.

- Table design text column:

Try not to use `**text**` column; because it affects performance. Use **varchar**() with a limit instead. Only use `text` column if it's really necessary, such as storing raw shipping label data, raw image data..

- For SQL statement, when using GROUP_CONCAT it's highly recommended to use | as separator. Because by default, most database use comma as a separator. When raw data is used for csv output; column cannot contain comma.
  For example, instead of
  SELECT GROUP_CONCAT(order_id) -- which yield 3,5,7 -> breaks the CSV download
  we should do
  SELECT GROUP_CONCAT(order_id SEPARATOR '|') -- which yields 3|5|7 -> does not break for the CSV download

- For SQL statements, do NOT put column on its own line like this

```
SELECT
    qa_id,
    MAX(invoice_item_id) AS example_invoice_item_id,
    MAX(invoice_id) AS invoice_id,
    MAX(si_num) AS si_num,
```

  because it takes forever to scroll up and down.

  Write columns on the same line like this:

```
SELECT
    qa_id, MAX(invoice_item_id) AS example_invoice_item_id, MAX(invoice_id) AS invoice_id,
MAX(si_num) AS si_num,
```

- SQL CASE:

For simple IF/ELSE, do NOT use CASE like this

```
CASE WHEN soi.product_notes IS NULL OR soi.product_notes = '' THEN NULL ELSE soi.product_notes END
```

Instead, is IF (if MSSQL, use IIF):

```
IF(soi.product_notes IS NULL OR soi.product_notes = '',NULL,soi.product_notes)
```

Only use CASE WHEN if there are more than one if/else branches.

- To check if a value exists, use if ($value !== null)
  Do not use if ($value)
  because when $value is zero, if ($value) does not pass.
  Example code:

```
if($quote->getIsMultiShipping() && $quoteItem->getCustomPrice()) {
    $item->setPrice($quoteItem->getCustomPrice())-
>setBaseOriginalPrice($quoteItem->getCustomPrice());
```

  This code is wrong. When customPrice is zero (no-charge), this if statement does not pass.

  Rewrite it to this:

```
if($quote->getIsMultiShipping() && ($quoteItem->getCustomPrice() !== null)) {
    $item->setPrice($quoteItem->getCustomPrice())-
>setBaseOriginalPrice($quoteItem->getCustomPrice());
```

now, if customPrice is zero, this if statement passes

- Class overriding/extending
  When overriding or extending a class; make sure to document at the top of the PHP/Javascript file. For example:

```
<?php
/**
 * Frans overrides Techgroup\UspsValidation\Controller\Index\Address
 */
```

- Javascript

Each function must have documentation explaining its purpose, and one or a few usage example(s)

```
/**
 * Frans Address Validation module for BE
 * 2024/09/02 Brian created
 * Usage: define('addrval', addrVal)
 *  addrVal.validateAddress
 */
define([
    'jquery',
    'ko',
    'Magento_Ui/js/form/form',
    'Magento_Ui/js/modal/modal',
    'Frans_Framework/frans/framework/axios.min',
], function ($, ko, Component, modal, axios) {
    'use strict';
```

- **Branching rules:**
When you work on a task, such as S1234, create a branch with the same name, such as branch s1234

For example, task https://plan.socalappsolutions.com/view.php?id=S1904p4 --> branch name = s1904p4
Notice that git does not work well if we name the branch with numbers only; so do NOT name it with numbers only, such as `1904`. Sometimes it works, sometimes git will throw an error because it thinks 1904 is a commit sha, not a branch name.

Make sure branches are lowercase, not uppercase. For example, branch name = s1904, not **S**1904

- PHP code to get table name: no need to call functions to get table name, just use string literal. Instead of

```
resourceConnection->getTableName('frans_order');
```

Just type ‘frans_order’

Revision #16
Created 21 January 2025 00:07:20 by Brian
Updated 25 March 2026 04:25:54 by Brian