**การใช้งานโปรแกรมและทำความรู้จักแต่ละฟิลด์**

**Bonanza Financial Market**

**เริ่มต้นการใช้งาน**

1. เปิดโปรแกรม
2. เลือก Hamburger Menu ![A blue background with white lines

   Description automatically generated](data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAADIAAAA2CAYAAACFrsqnAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAB+SURBVGhD7dixDYAwDAVRw5hIMAdMwB5IsCY0LlJHKQ7rXhO3V/wm07rfbxQw5/t7htAYQmMIjSE0htAYQmMIjSE0htAYQlMmpOs76DqXvMbYjievfv5r0RhC49hp3AiNYx/JsTfcCI0hNIbQGEJjCI0hNIbQGEJjCE2RkIgP4nIatUM2luoAAAAASUVORK5CYII=) ด้านซ้ายบนของโปรแกรม
3. กดเลือกเมนูหัวข้อ Interface  🡪 Bonanza 🡪 Template

**เมนู Template**

เมื่อเข้ามาสู่เมนู Template เมนูจะแสดงตารางข้อมูลต่างๆที่ผู้ใช้งานต้องการ รวมไปถึง เครื่องมือต่างๆที่สามารถใช้งานได้

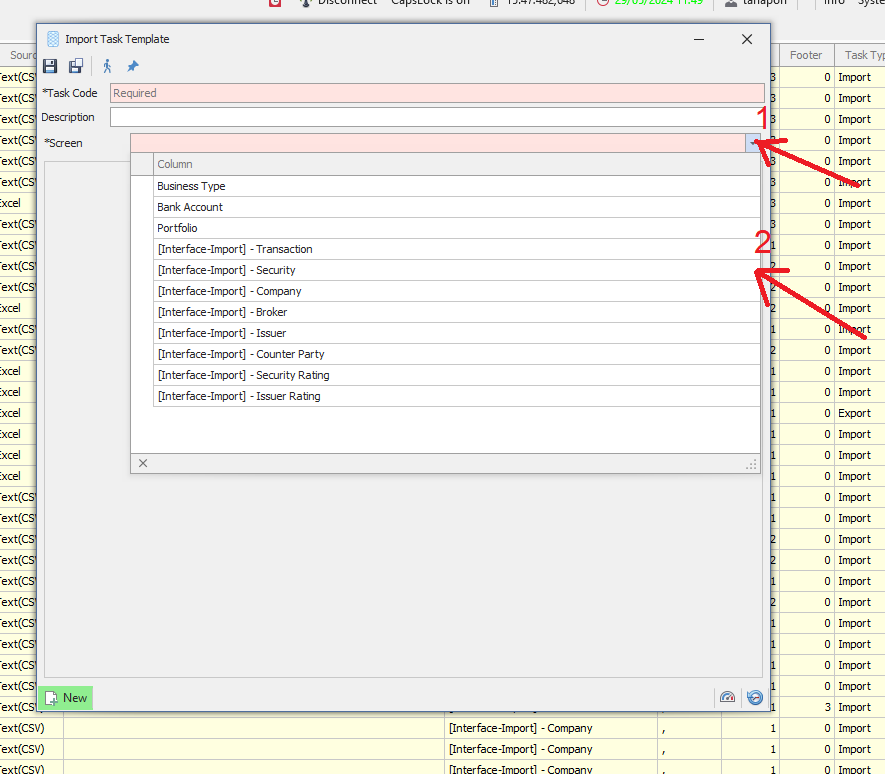
1. กดปุ่ม ดังรูป

A screenshot of a computer

Description automatically generated

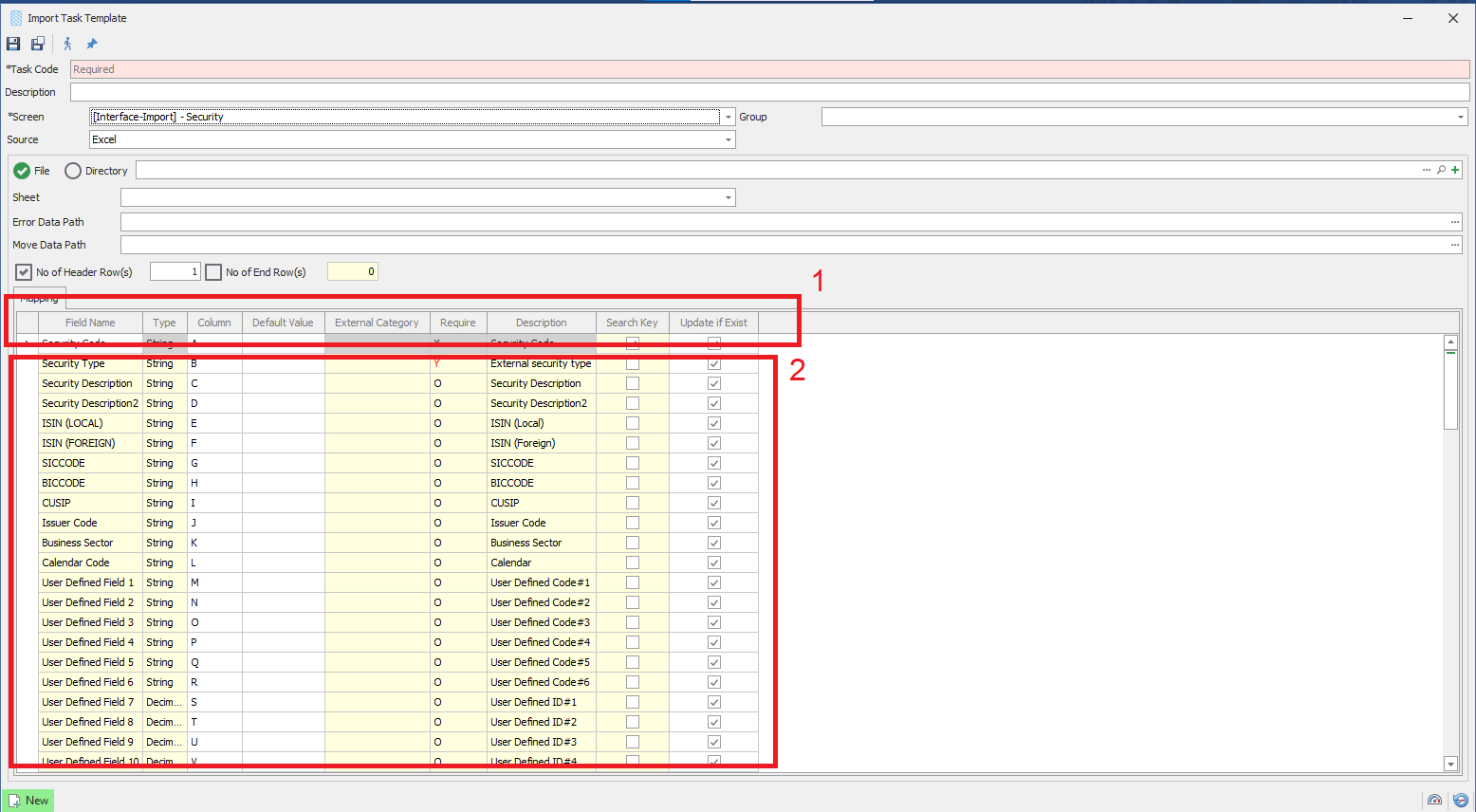
1. A screenshot of a computer

   Description automatically generatedเลือก Import
2. กดเลือกข้อมูล และกดปุ่มตามลำดับ



**ทำความรู้จักฟิลด์ในแต่ละส่วน**

คอลัมน์และข้อมูลต่างใน Template



1.คอลัมน์

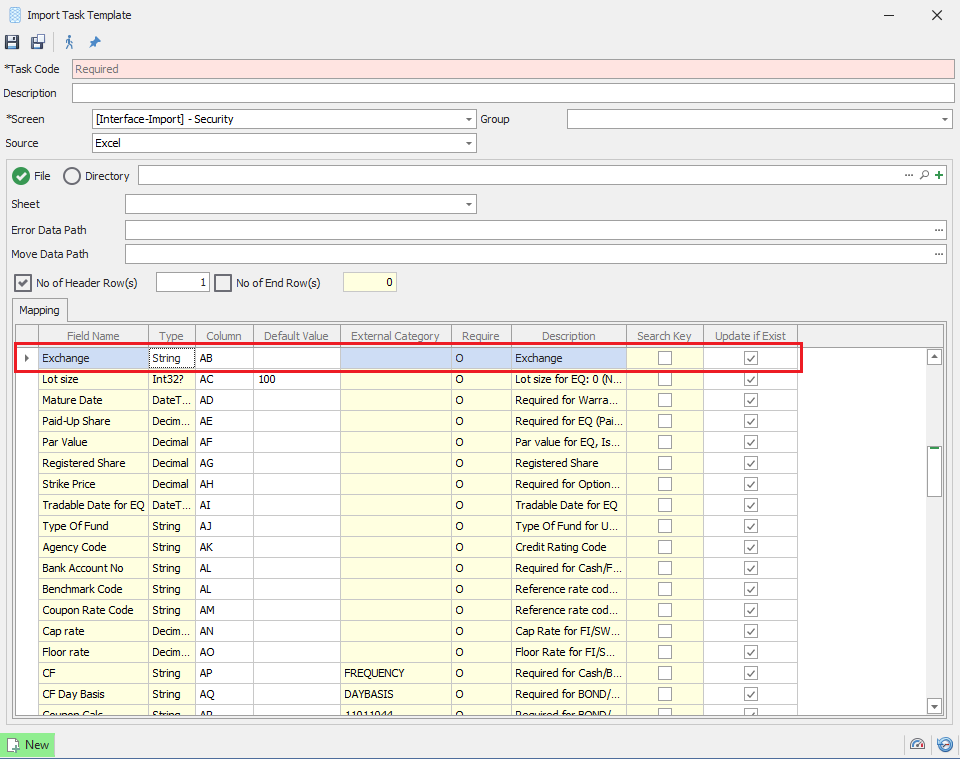
* Field Name ชื่อฟิลด์ในคอลัมน์
* Type ประเภทของข้อมูล จะมี String Decimal DateTime Int32 Int16 Char
* Column การเรียงลำดับข้อมูลในคอลัมน์
* Default Value ค่าเริ่มต้นข้อมูล
* External Category หมวดหมู่ข้อมูล
* Require
* Description คำอธิบายข้อมูล
* Search Key
* Update if Exist

2.ข้อมูลที่มีอยู่ในตาราง

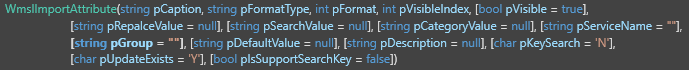
**ส่วนของโค้ดที่เกี่ยวกับข้องกับฟิลด์**

จะขออธิบายเกี่ยวความสัมพันธ์ในโค้ด โดยในส่วนของหน้า **Import Task Template** จะสามารถหาและศึกษาเกี่ยวกับโปรแกรมหน้านี้ได้จาก ไฟล์โค้ด SecurityDTO.cs และ ImportSecurityDSDTO.cs

โดยในตัวอย่างนี้จะยกตัวอย่างข้อมูลในคอลัมน์ **Field Name “**Exchange**”** โดยจะเป็นการนำเข้าข้อมูลในโปรแกรม



* ขอเพิ่มเติม สิ่งที่ควรรู้เกี่ยวกับการเก็บตัวแปรต่างๆไว้เราควรทำความรู้จักกับตัวแปรก่อนที่จะรู้จักข้อมูลในส่วนของคอลัมน์โดยบางคอลัมน์อาจจะไม่แสดงมาเป็นฟิลด์ให้เห็นแต่จะสามารถยังทำงานได้อยู่

**ตัวแปรที่ควรรู้ จากภาพและตารางข้างต้น**

**ตาราง ความสัมพันธ์ของตัวแปรและ Field**

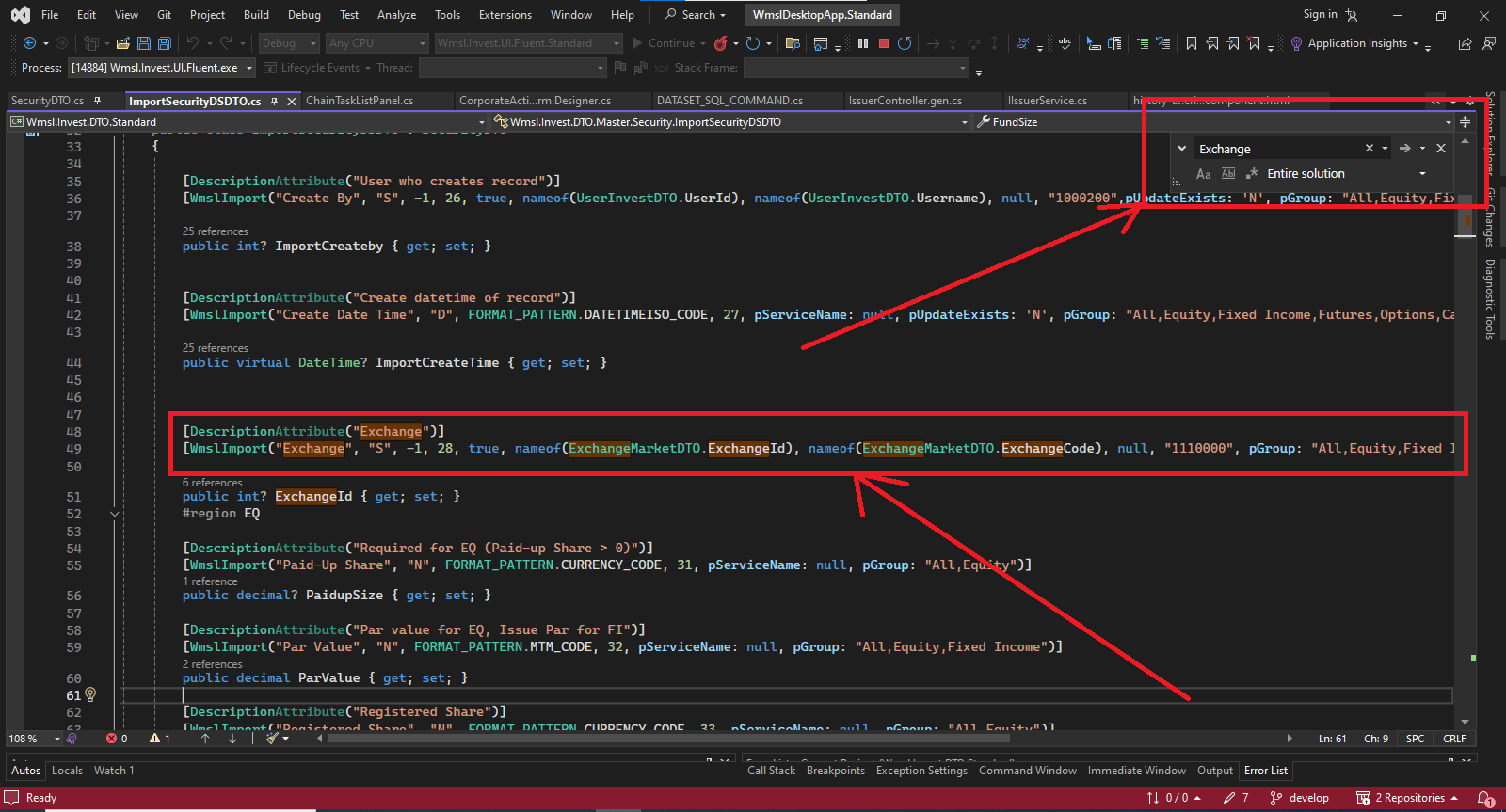
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **variable** | **Type** | **Field** | **Detail** |
| 1 | pCaption | String | Field Name | ใช้เก็บชื่อของคอลัมน์นั้นที่ต้องการ แทนในฟิลด์ด้วย |
| 2 | pFormatType | String | Type | ใช้เก็บ Type (ประเภทของข้อมูล จะมี String Decimal DateTime Int32 Int16 Char) แทนในฟิลด์ด้วย “Field Name” |
| 3 | pFormat | Int |  | ใช้ระบุค่า ประเภทการระบุข้อมูล โดยจะมี S , N , D และ  - S จะ แทนด้วย -1 ที่จะเป็นการระบุ String , - N จะแทนด้วยFORMAT\_PATTERN.CURRENCY\_CODE เป็น การกำหนดรูปแบบการใส่ข้อมูลเป็นตัวเลข เช่น กำหนดตัวเลขระหว่าง 1-10 หรือ จำกัดจำนวน 1 และ 2  - D จะแทนด้วย FORMAT\_PATTERN.DATEISO\_CODE เป็นการกำหนดรูปแบบการใส่ข้อมูลเป็นวันที่ โดยขึ้นอยู่กับการกำหนด เช่น 10/1/2000 |
| 4 | pVisibleIndex | Int | Column | ใช้การแสดงผลในคอลัมน์บนหน้าจอ เพื่อให้ผู้ใช้งานเห็นข้อมูลตามลำดับที่ต้องการ |
| 5 | pVisible | Bool |  | ใช้กำหนดว่าคุณสมบัติ (property) นั้นสามารถมองเห็นผ่าน UI (true/ false) |
| 6 | pRepalceValue | String |  | แทนที่ค่าที่ถูกระบุมาจาก pSearchValue ตาม id หรือ name ที่สอดคล้องกัน แทนในฟิลด์ด้วย |
| 7 | pSearchValue | String |  | ใช้ค้นหาที่ถูกระบุมาโดยจะไปค้นหาในฐานข้อมูลหลังจากนั้นจะส่งไปเพื่อหาจาก pRepalceValue |
| 8 | pCategoryValue | String | External Category | ใช้เก็บค่าหมวดหมู่ของฟิลด์ ให้เป็นไปตามเงื่อนไข |
| 9 | pServiceName | String |  | เป็นการเรียกใช้ ServiceId เพื่อดำเนินการ |
| 10 | pGroup | String |  | ใช้กำหนดกลุ่มของฟิลด์ เพื่อจัดระเบียบข้อมูลในกลุ่มเดียวกัน โดยจะต้องระบุด้วย key ที่ถูกประกาศไว้ |
| 11 | pDefaultValue | String | Default Value | ใช้เพื่อกำหนดค่าเริ่มต้นสำหรับฟิลด์ เมื่อไม่มีข้อมูลถูกระบุในระหว่างการอัปโหลดหรือการสร้างข้อมูลใหม่ |
| 12 | pDescription | String | Description | ใช้เก็บอธิบายความสามารถหรือความหมายของฟิลด์นั้นๆ |
| 13 | pKeySearch | char | Search Key | ใช้ระบุว่าคุณสมบัติของ property นั้นรองรับการค้นหาด้วยคีย์หรือไม่ โดยค่า 'Y' หมายถึงใช่ (Yes) และ 'N' หมายถึงไม่ (No) |
| 14 | pUpdateExists | char | Update if Exist | ระบุว่าในกรณีที่ข้อมูลนี้มีการอัปเดตอยู่แล้ว ต้องทำการอัปเดตข้อมูลหรือไม่ โดยค่า 'Y' หมายถึงใช่ (Yes) และ 'N' หมายถึงไม่ (No) |
| 15 | pIsSupportSearchKey | Bool |  | ใช้การกำหนดว่าคุณสมบัติของ property นั้น ๆ รองรับการค้นหาด้วยคีย์หรือไม่ |
| 16 | Require | - | - | ใช้สำหรับการกำหนด ฟังก์ชั่น ถ้ามีการกำหนด IsRequiredAttribute จะแสดง “Y” แต่ถ้าไม่กำหนดจะแสดง “O” |

* เมื่อรู้จักตัวแปรต่างๆแล้ว ก็สามารถรู้การส่งและรับของข้อมูลและการทำงานของโปรแกรมได้ โดยจะมี วิธีดังนี้

เปิดไฟล์ที่ ตามชื่อ Field Name ที่ต้องการ ในตัวอย่างนี้จะใช้ Field Name ชื่อ “Exchange”

A screenshot of a computer

Description automatically generated

* ทำการค้นหาตามชื่อ Field Name ในที่นี้คือ “**Exchange**”
* ในส่วนนี้คือ โค้ดที่ใช้ระบุข้อมูลลงคอลัมน์ในโปรแกรม
* จากโค้ดดังกล่าวจะสามารถนำส่งข้อมูลเข้าสู่ตารางข้อมูลได้ดังนี้



เหตุผลที่ข้อมูลออกมาในรูปนี้ก็เพราะ จากการกำหนด Constructor ที่มีการเก็บตัวแปรที่สามารถเก็บค่าของข้อมูลได้ จาก “**ตาราง ความสัมพันธ์ของตัวแปรและ Field”**ดังนั้นจะสามารถระบุข้อมูลลงตารางได้ดังนี้

1.pCaption จะเป็นข้อมูลเป็นประเภท String ที่มาจาก **Field Name** “Exchange”

2.pFormatType จะเป็นข้อมูลเป็นประเภท String ที่มาจาก **Type**  “S” = “String”

3.pFormat จะเก็บข้อมูลเป็นประเภท int ที่มาจาก (-1) หรือคือการใส่ข้อมูลเป็น String

4. pVisibleIndex จะเก็บข้อมูลเป็นประเภท int ที่มาจาก **Column** (28) = AB

5. pVisible จะเก็บข้อมูลเป็นประเภท Boolean ที่มาจาก (true)

6.ขั้นตอนการส่งและการแสดงข้อมูลในตารางโดยการติดต่อไปยังฐานข้อมูล

6.1 pRepalceValue จะเก็บข้อมูลเป็น String ที่มาจาก nameof(ExchangeMarketDTO.ExchangeId)

6.2 pSearchValue จะเก็บข้อมูลเป็น String ที่มาจาก nameof(ExchangeMarketDTO.ExchangeCode)

6.3 pCategoryValue จะเก็บข้อมูลเป็น String ที่มาจาก **External Category** ในที่นี้จะเป็นค่า(null)

6.4 pServiceName จะเก็บข้อมูลเป็น String ที่มาจาก ServiceId (1110000)

เมื่อทราบถึงการเก็บตัวแปรแล้วจะเข้าสู่ การทำงานในขั้นตอนของการเชื่อมโยงข้อมูลและนำมาแสดงผล ดังนี้

6.4.1. ระบบทำการค้นหาข้อมูลในฐานข้อมูล ที่ตรงตามเงื่อนไข แต่ในตัวอย่างจะแสดงค่า null ทำให้ไม่ต้องผ่านเงื่อนไข

6.4.2. โดยใช้หมายเลข ServiceId คือ (1110000) ในการดำเนิน

6.4.3. เมื่อรู้ระบบส่งคำสั่งและการค้นหาของฐานข้อมูลมาแล้วจะทำการ ตรวจสอบว่ามี nameof(ExchangeMarketDTO.ExchangeCode) (จากตัวแปร pSearchValue) ที่ตรงกับใน nameof(ExchangeMarketDTO.ExchangeId) จากตัวแปร(pRepalceValue) หรือไม่

ถ้ามี จะรีเทิร์นกลับมาเป็น nameof(ExchangeMarketDTO.ExchangeId)



7. pGroup จะเก็บข้อมูลเป็น String ที่มาจาก "All,Equity,Fixed Income,Futures,Options,Cash,FX,Swap"



8. pDefaultValue จะเก็บข้อมูลเป็น String ที่มาจากการไม่ได้กำหนดค่าให้ตัวแปรทำให้เป็นค่าเริ่มต้นคือ null



9. pDescription จะเก็บข้อมูลเป็น String ที่มาจาก Description "Exchange"



10. pKeySearch จะเก็บข้อมูลเป็น char ที่มาจาก Search Key กำหนดเป็นค่าเริ่มต้น คือ N

A screenshot of a computer

Description automatically generated

* กรณี มีการมีการกำหนดจะให้เป็นไปตามค่าที่กำหนด เช่น กำหนดค่าเป็นY ก็จะเป็นดังรูป

A screenshot of a computer

Description automatically generated

11. pUpdateExists จะเก็บข้อมูลเป็น char ที่มาจาก Update if Exist กำหนดเป็นค่าเริ่มต้น คือ Y

A screenshot of a computer

Description automatically generated

* กรณี มีการกำหนดจะให้เป็นไปตามค่าที่กำหนด เช่น กำหนดค่าเป็น N



12. Require จะขึ้นอยู่กับการกำหนด ฟังก์ชั่น IsRequiredAttribute ในโค้ดดังกล่าวไม่มีการกำหนดจะทำให้เป็น O

A close up of a box

Description automatically generated

* กรณี มีการกำหนด ฟังก์ชัน IsRequiredAttribute จะทำให้ค่าที่แสดงเป็น Y



**การเพิ่มและลบฟังก์ชั่นของปุ่มและคอลัมน์**

ในส่วนของโค้ดจะมีการกำหนดปุ่มฟังก์ชันบางฟังก์ชันเพื่อที่จะใช้งานในการ Import และ export หรือการปิดเปิดการใช้งานของปุ่มบางปุ่มในโปรแกรม โดยจะมีการสังเกตและการใช้งานดังนี้ โดยจะยกตัวอย่างดังนี้

* **หน้าหลัก Issuer**A screenshot of a computer

  Description automatically generated

1. การที่จะแสดงให้เห็นถึงปุ่ม Import จะต้องประกาศฟังก์ชั่น ISupportImport เพื่อที่จะแสดงปุ่มดังกล่าว กรณีอยากจะปิดการใช้งานจะต้องลบฟังก์ชั่น ISupportImport ออก
2. การที่จะแสดงให้เห็นถึงปุ่ม Export จะต้องประกาศฟังก์ชั่น ISupportExportเพื่อที่จะแสดงปุ่มดังกล่าว

กรณีอยากจะปิดการใช้งานจะต้องลบฟังก์ชั่น ISupportExport ออก

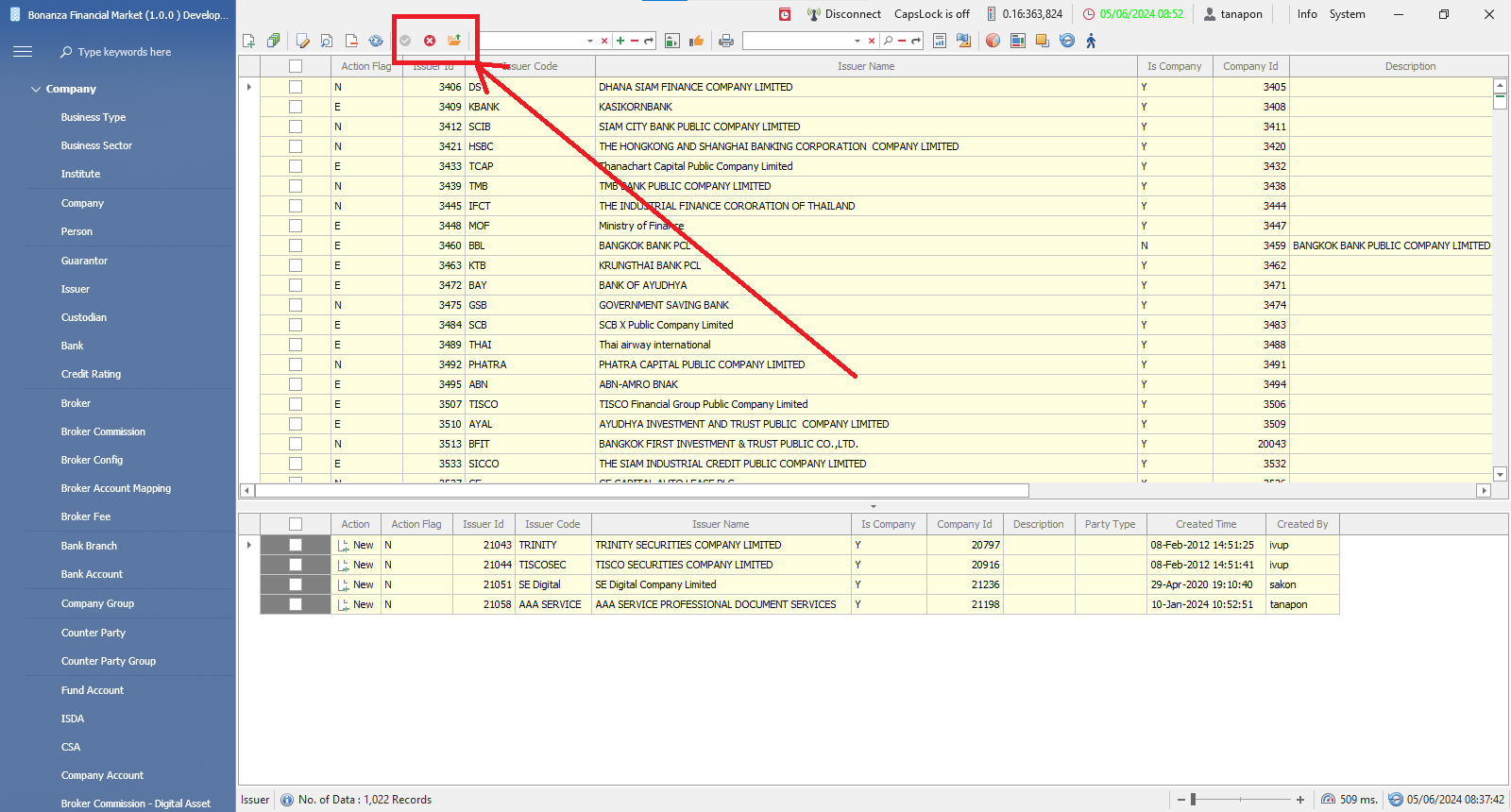
* A screenshot of a computer

  Description automatically generated**ตัวอย่างการใส่ฟังก์ชันในโค้ด (**ISupportImport**) , (**ISupportExport**)**

)

โดยจะต้องระบุฟังก์ชั่นที่ต้องไว้ หลังจาก public class ด้วย ชื่อฟังก์ชั่น ดังรูปA computer screen with text on it

Description automatically generated

กรณีที่จะปิดการทำงาน ของปุ่ม import เราก็ต้องลบฟังก์ชั่น (ISupportImport) และจะแสดงผล ดังรูป

* **ตัวอย่างการใส่ฟังก์ชันในโค้ด** (ISupportExport)A computer screen shot of a black background

  Description automatically generated

กรณีที่จะปิดการทำงาน ของปุ่ม Export เราก็ต้องลบฟังก์ชั่น (ISupportExport) และจะแสดงผล ดังรูป

A screenshot of a computer

Description automatically generated

* **ตัวอย่างการใส่ฟังก์ชันในโค้ด** (ISupportImport) A computer screen with text

  Description automatically generated

นอกจากนี้ยังมีฟังก์ชั่นที่สามารถควบคุมการทำงานของข้อมูลได้ ยกตัวอย่างเช่น INotSupportUpdateExists ที่จะปิดการทำงานคอลัมน์ ในโปรแกรมจะแสดงดังนี้

**กรณีที่ไม่ได้ใส่ฟังก์ชั่น**

A screenshot of a computer

Description automatically generated

**กรณีใส่ฟังก์ชั่น INotSupportUpdateExists**

A screenshot of a computer

Description automatically generated

**กรณีใส่ฟังก์ชั่น INotSupportSearchKey**

A screenshot of a computer

Description automatically generated