VMTest

[Type your address] ⦁ [Type your phone number] ⦁ [Type your e-mail address]

iunittest

Using iUnittest ⦁ September 2010 ⦁ by Prazit ® Jitmanozot

***สารบัญ***

**ใช้งานครั้งแรก**

* สิ่งที่ต้องมีก่อนการใช้งาน
* ตั้งค่า SQLCA Connection
* ตั้งค่า SQLUT Connection

**เริ่มทดสอบฟังชั่น**

* เลือกฟังชั่นที่ต้องการทดสอบจากเมนู
* ป้อนค่าให้กับ Arguments ของฟังชั่น
  + ค่าที่เป็น String
  + การเรียกดูค่าที่เป็น String หลายบรรทัด
  + ค่าที่เป็น Integer,Long และ Decimal
  + ค่าที่เป็น Date
  + ค่าที่เป็น Time
  + ค่าที่เป็น Datetime
  + ค่าที่เป็น DatawindowXML
  + การเรียกดูค่าที่เป็น DatawidowXML ใน Datawindow
  + ค่าที่เป็น Boolean
* กำหนดใช้ DebugMode
* เรียกฟังชั่นให้ทำงาน
* ข้อผิดพลาดที่คาดไม่ถึง
* สั่งหยุดประมวลผล
* ผลลัพธ์จากฟังชั่น
  + Multiline Viewer
  + DatawindowXML Viewer

**เทสเคส**

* บันทึกค่าที่ป้อนเก็บไว้ในเทสเคส
* ดึงค่าในเทสเคสออกมาใช้
* ลบค่าในเทสเคส
* ปรับปรุงเทสเคสเดิม

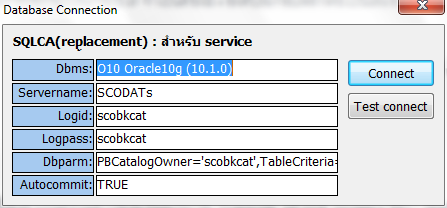
**ใช้งานครั้งแรก**

ก่อนการใช้งานโปรแกรมนี้ได้ จำเป็นต้องมี 4 องค์ประกอบหลักต่อไปนี้เสมอ

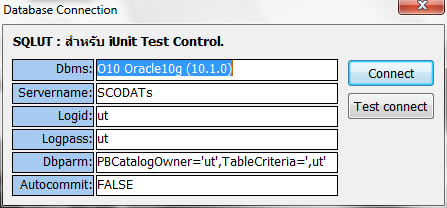
1. โปรแกรม iUnittest (source code)
2. โปรแกรม Powerbuilder 12 beta 2(build5107) ซึ่งจะเป็น portable หรือไม่ก็ได้ (จากนี้จะขอเรียกว่า PB)
3. ฐานข้อมูล iUnittest (จากนี้จะขอเรียกว่า SQLUT)
4. ฐานข้อมูลสหกรณ์ (จากนี้จอขอเรียกว่า SQLCAreplacement)

เมื่อมีสิ่งจำเป็นครบแล้วให้เริ่มรันโปรแกรม Powerbuilder แล้วเปิด Workspace ชื่อว่า PBService ซึ่งปกติจะอยู่ที่ตำแหน่ง C:\GCOOP\PBSERVICE\ แล้วเลือก Run Target ที่ชื่อว่า iUnittest

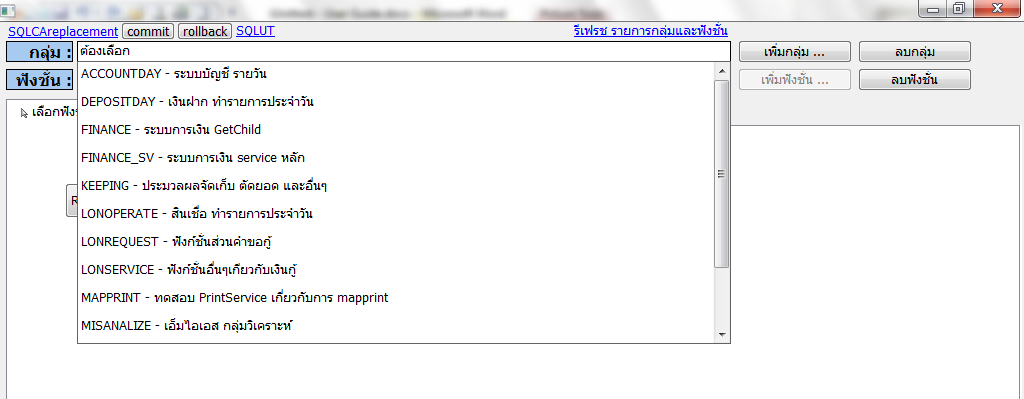
หากการตั้งค่าการเชื่อมต่อฐานข้อมูลยังไม่ถูกต้อง โปรแกรมจะเปิดหน้าต่าง Database Connection ขึ้นมาให้ตั้งค่า SQLCA(replacement) Connection กับ SQLUT Connection ตามลำดับ ซึ่งจำเป็นต้องตั้งค่าให้ถูกต้องก่อนคลิกที่ปุ่ม Connect เพื่อนำค่าไปใช้ (ควรคลิกปุ่ม Test Connect ให้แน่ใจก่อนว่าถูกต้องแล้ว)



***TGI001: ตั้งค่า SQLCAreplacement Connection (SQLCAreplacement จะถูกใช้จากฟังชั่น)***



***TGI002: ตั้งค่า SQLUT Connection (SQLUT จะถูกใช้จากตัวโปรแกรม iUnittest เอง)***

****

***TGI003: หน้าจอแรกของโปรแกรม iUnittest (หลังการตั้งค่า Database Connection สมบูรณ์แล้ว)***

**เริ่มทดสอบฟังชั่น**

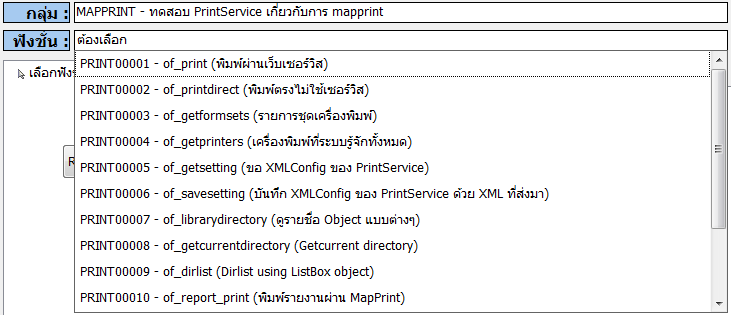
ลำดับการทดสอบฟังชั่นหนึ่งๆจะเป็นไปตามลำดับต่อไปนี้เสมอ

* 1. เลือกฟังชั่นที่ต้องการทดสอบจากเมนู
  2. ป้อนค่าให้กับ Arguments ของฟังชั่น
  3. กำหนดใช้/ไม่ใช้ DebugMode
  4. เรียกฟังชั่นให้ทำงาน
  5. ข้อผิดพลาดที่คาดไม่ถึง
  6. ผลลัพธ์จากฟังชั่น

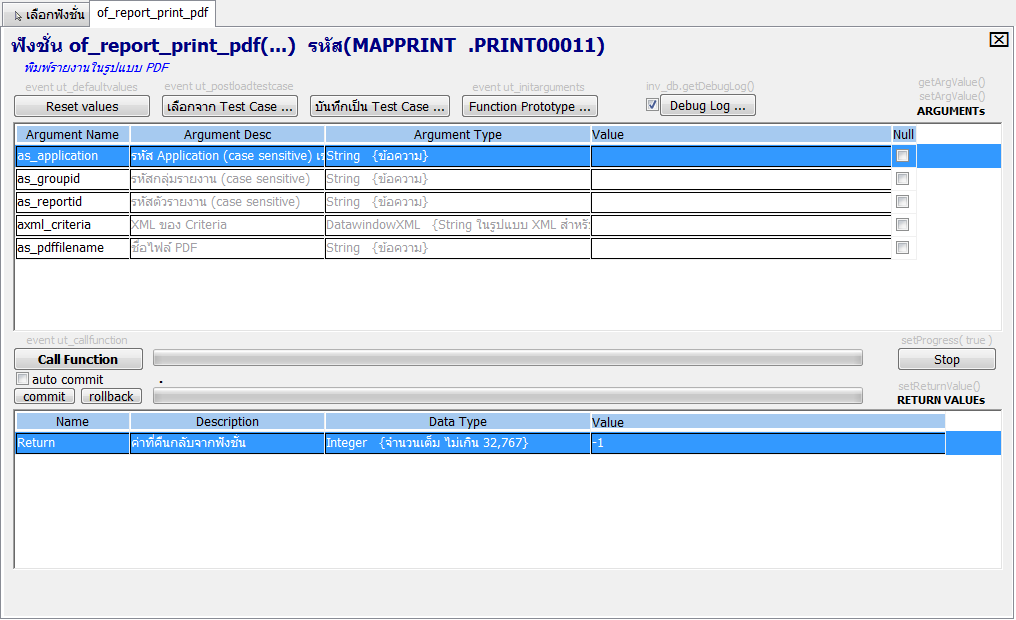


***TGI004: บริเวณเมนูขณะที่ยังไม่ได้เลือกฟังชั่นที่ต้องการ***

1. **เลือกฟังชั่นที่ต้องการทดสอบจากเมนู** ให้เริ่มคลิกที่กลุ่มจะมี DROPDOWN แสดงรายชื่อกลุ่มขึ้นมา(ดังภาพ TGI003)จากนั้นคลิกที่ชื่อกลุ่มที่ต้องการจะมี DROPDOWN แสดงรายชื่อฟังชั่นในกลุ่มที่เลือกขึ้นมา(ดังภาพ TGI005)จากนั้นให้คลิกที่ชื่อฟังชั่นที่ต้องการทดสอบจะปรากฏแท็บของฟังชั่นขึ้นที่ส่วนล่างของโปรแกรม(ดังภาพ TGI006)



***TGI005: รายชื่อฟังชั่นในกลุ่มแสดงขึ้นหลังจากคลิกที่ชื่อกลุ่ม***



***TGI006: แท็บของฟังชั่นแสดงขึ้นหลังจากคลิกที่ชื่อฟังชั่น***

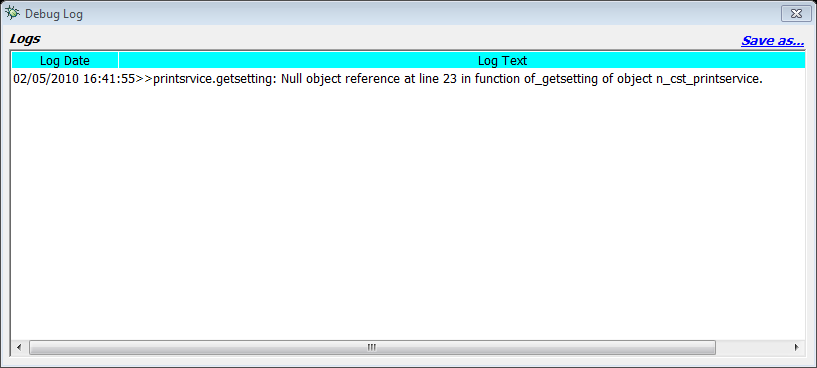
1. **ป้อนค่าให้กับ Arguments ของฟังชั่น** ให้เริ่มดูจากรูป TGI006 สังเกตุที่บริเวณกรอบสีแดงจะเป็นช่องที่ให้ป้อนค่าสำหรับ Arguments แต่ละตัว (จากนี้จะเรียกว่า ช่องกรอกข้อมูล) ซึ่ง 1 บรรทัดหมายถึง 1 Argument มีรายละเอียดเกี่ยวกับ Argument นั้นแสดงอยู่ด้านซ้ายมือของกรอบสีแดง จุดสำคัญคือ Argument Type เป็นตัวบอก Datatype ของ Argument นั้น ซึ่งมีการป้อนค่าที่แตกต่างกันสำหรับแต่ละ Datatype ดังนี้

* ค่าที่เป็น **String** – สามารถป้อนได้ไม่จำกัดความยาวและจำนวนบรรทัดโดยไม่ต้องครอบด้วยเครื่องหมายคำพูด
  + การเรียกดูค่าที่เป็น String หลายบรรทัด สามารถทำได้โดยการดับเบิ้ลคลิกที่ช่องกรอกข้อมูล หรือคลิก Mouse ขวาที่ช่องกรอกข้อมูลแล้วคลิกที่ **Show value in Multiline Viewer …** *{ข้อมูลเพิ่มเติมดูที่หัวข้อ Multiline Viewer}*
* ค่าที่เป็น **Integer**,**Long** และ **Decimal** – ให้ป้อนชุดตัวเลขพร้อมจุดทศนิยมได้โดยไม่ต้องมีตัวคั่นหลักพันและหลักล้าน
* ค่าที่เป็น **Date** – ให้ป้อนตามรูปแบบ YYYY-MM-DD เช่น 2010-12-30
* ค่าที่เป็น **Time** – ให้ป้อนตามรูปแบบ hh:mm:ss เช่น 24:59:59
* ค่าที่เป็น **Datetime** – ให้ป้อนตามรูปแบบ YYYY-MM-DD หรือ YYYY-MM-DD hh:mm:ss
* ค่าที่เป็น **DatawindowXML** – สามารถป้อนได้ไม่จำกัดความยาวและจำนวนบรรทัดโดยไม่ต้องครอบด้วยเครื่องหมายคำพูดแต่ต้องอยู่ในรูปแบบของ XML สำหรับ Datawindow เท่านั้น
  + การเรียกดูค่าที่เป็น DatawidowXML ใน Datawindow สามารถทำได้โดยการคลิก Mouse ขวาที่ช่องกรอกข้อมูลแล้วคลิกที่ **Show value in DatawindowXML Viewer …** *{ข้อมูลเพิ่มเติมดูที่หัวข้อ DatawindowXML Viewer}*
* ค่าที่เป็น **Boolean** – ไม่สามารถกำหนดใช้งานได้ ให้เปลี่ยนไปใช้ Integer แทนโดยกำหนดให้ค่า TRUE มีค่า =1 และค่า FALSE มีค่า =0 และเขียน Script ที่ Unittest เพื่อแปลงค่าตามนี้ *{ข้อมูลเพิ่มเติมดูที่ Developer Guide หัวข้อ getArgument }*

1. **กำหนดใช้/ไม่ใช้ DebugMode** ให้สังเกตุที่บริเวณเหนือช่องกรอกข้อมูลจะมีกล่องตัวเลือก DebugMode อยู่ข้างซ้ายปุ่ม Debug Logหากใส่เครื่องหมายถูกไว้จะหมายถึงให้เปิดการทำงานของ Debug Mode ซึ่งจะทำให้ปุ่ม Debug Log สามารถคลิกได้และที่ปุ่ม Debug Log นี้เอาไว้เรียกดู หรือ SaveAs Log ทั้งหมดที่ฟังชั่นทำการบันทึกเอาไว้ได้ *{ข้อมูลเพิ่มเติมดูที่ Developer Guide หัวข้อ Debug Log }*

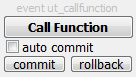
****

***TGI007: กล่อง CheckBox สำหรับ DebugMode และปุ่ม Debug Log***

******

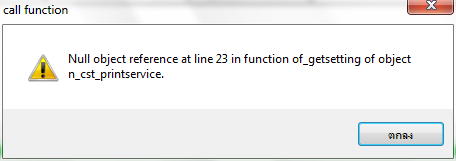
***TGI008: หน้าต่าง Debug Log หลังคลิกที่ปุ่ม Debug Log…***

1. **เรียกฟังชั่นให้ทำงาน** ให้สังเกตุทางด้านซ้ายมือบริเวณใต้กรอบ Arguments จะพบปุ่ม Call Function และตัวเลือกAuto Commitอยู่ใต้ปุ่มซึ่งโดยปกติสามารถคลิกที่ปุ่ม Call Function เพื่อเรียกฟังชั่นให้ทำงานได้ทันที แต่ถ้าหากต้องการให้โปรแกรมทำการ Commit ให้อัตโนมัติหลังจากฟั่งชั่นทำงานเสร็จเรียบร้อยแล้วก็ให้ใส่เครื่องหมายถูกไว้ที่ตัวเลือก Auto Commit ด้วย (การทำงานของปุ่ม COMMIT, ROLLBACK และ AUTO COMMIT จะกระทำไปที่ SQLCAreplacement เท่านั้น)



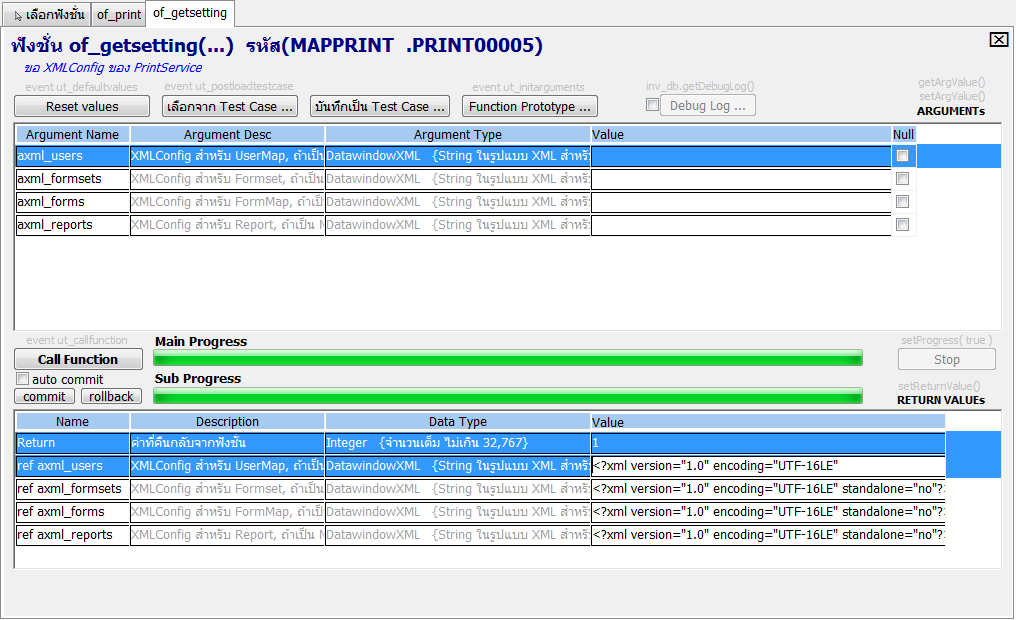
***TGI009: ปุ่ม Call Function และชุดตัวเลือก Auto Commit***

1. **ข้อผิดพลาดที่คาดไม่ถึง** จะถูกดักจับโดยโปรแกรม iUnittest อย่างอัตโนมัติ และจะ Copy ข้อความแจ้งไว้ใน Clipboard พร้อมกับแสดงที่หน้าจอด้วย MessageBox ทันทีที่พบข้อผิดพลาดนั้นๆ ซึ่งสามารถคลิกตอบ ตกลง/OK เพื่อปิด MessageBox แล้วทดสอบต่อได้ทันทีโดยไม่ทำให้โปรแกรมตาย ไม่ต้องรันโปรแกรมใหม่ (มีประโยชน์มากกรณีที่ต้องการทดสอบเทสเคสหลายๆแบบบนฟังชั่นเดียวกัน)

****

***TGI010: ตัวอย่าง MessageBox แสดงข้อผิดพลาดที่คาดไม่ถึง***

1. **สั่งหยุดประมวลผล** สำหรับฟังชั่นที่เป็นการประมวลผลจะมีการแสดงความคืบหน้าให้เห็นที่หน้าจอได้บริเวณ Main Progress และ Sub Progress อยู่ข้างปุ่ม Call Function ซึ่งขณะประมวลผลอยู่ก็สามารถสั่งให้หยุดการทำงานได้ทันที โดยคลิกที่ปุ่ม Stop ทางด้านขวาของ Main Progress



***TGI011: แท็บของฟังชั่นหลังคลิกปุ่ม CallFunction และได้ผลลัพธ์มาแสดงในส่วน Return Values***

1. **ผลลัพธ์จากฟังชั่น** ให้ดูที่รูป TGI011 ใต้ Sub Progress บริเวณกรอบสีแดง จะเป็นส่วนการแสดงค่าผลลัพธ์ที่คืนกลับมาจากฟังชั่น (**Return Values**) ซึ่งโปรแกรมจะสามารถแสดงค่าได้ทุก Type เช่นเดียวกับในส่วนของ **Arguments** และสามารถเรียกดูข้อมูลด้วย Multi Line Viewer และ DatawindowXML Viewer ได้เช่นกัน

**เทสเคส**

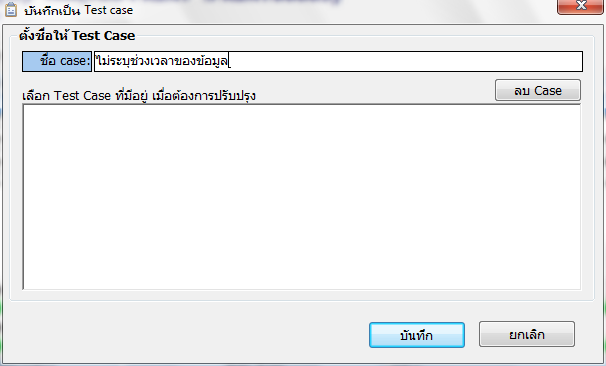
เทสเคสคือค่าของ Arguments ที่เราป้อนและบันทึกเก็บเอาไว้ ซึ่งสามารถเรียกค่าชุดนั้นกลับขึ้นมาใช้ได้ทันทีโดยไม่ต้องป้อนใหม่ และสามารถบันทึกค่าที่แตกต่างกันเก็บไว้ได้แบบไม่จำกัดจำนวน เพื่อให้สามารถเรียกดูผลลัพธ์ของฟังชั่นในกรณีต่างๆกันได้ตลอดเวลา

****

***TGI012: ปุ่มบันทึก Test Case และเลือก Test Case***

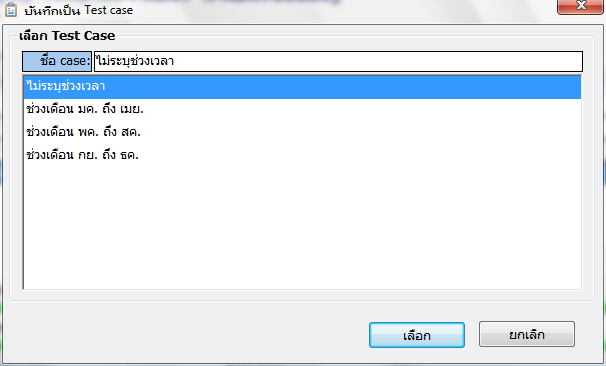
1. **บันทึกค่าที่ป้อนเก็บไว้ในเทสเคส** ให้คลิกที่ปุ่ม บันทึกเป็น Test Case (ตามรูป TGI012) จะมีหน้าต่างบันทึก Test Case แสดงขึ้นมาเพื่อให้ตั้งชื่อให้กับชุดค่าข้อมูลที่กำลังบันทึกและจะเรียกข้อมูลชุดนี้ว่า Case (เคส)

ให้ตั้งชื่อเคสได้ตามต้องการแต่จะยาวเกินกว่า 200 ตัวอักษรไม่ได้ (ซึ่งปกติไม่ควรยาวเกินขนาดช่องกรอก) เมื่อเสร็จแล้วให้คลิกที่ปุ่มบันทึกเพื่อเก็บค่าเอาไว้ใช้ต่อไป



***TGI013: หน้าต่างบันทึก Test Case (ตัวอย่างนี้ตั้งชื่อ case ว่า ไม่ระบุช่วงเวลา)***

1. **ดึงค่าในเทสเคสออกมาใช้** ให้คลิกที่ปุ่ม เลือกจาก Test Case… (ตามรูป TGI012) จะมีหน้าต่างเลือกเทสเคสแสดงขึ้น ให้เลื่อนเมาท์ไปคลิกที่ชื่อเคสที่ต้องการแล้วคลิกที่ปุ่มเลือก หน้าต่างเลือกเทสเคสจะหายไปพร้อมกับป้อนค่าข้อมูลจากเคสนั้นลงใน Arguments ให้โดยอัตโนมัติ



***TGI014: หน้าต่างเลือก Test Case***

1. **ลบค่าในเทสเคส** ให้คลิกที่ปุ่ม บันทึกเป็น Test Case… เมื่อหน้าต่างบันทึกเทสเคสแสดงขึ้น ให้เลื่อนเมาท์ไปคลิกที่ชื่อเคสที่ต้องการลบแล้วคลิกที่ปุ่ม ลบCase
2. **ปรับปรุงเทสเคสเดิม** ให้ใช้วิธีลบเคสเดิมทิ้งก่อนแล้วบันทึกชื่อเคสเดิมอีกครั้ง โดยให้คลิกที่ปุ่ม บันทึกเป็น Test Case… คลิกที่ชื่อเคสที่ต้องการปรับปรุงแล้วคัดลอกชื่อเคสไว้แล้วคลิกที่ ลบCase ให้คลิกที่ปุ่ม บันทึกเป็น Test Case อีกครั้งแล้ววางชื่อเคสที่คัดลอกไว้ลงที่ช่องกรอกชื่อเคสแล้วคลิกที่ปุ่ม บันทึก