

โครงการ: ทดสอบ Java Code ด้วย JUnit (Bank)						
รหัสสถานการณ์การทดสอบ: JUnit003			ผู้เขียนสถานการณ์การทดสอบ: อรณิชา ศรีสมาน , ฮาปีปะห์ มะแซ			
ฟังก์ชันที่ทดสอบ: เพิ่ม ลบ ผาก ถอน และดึงข้อมูลบัญชี			วันที่เขียน: 12 ตุลาคม 2567			
<p>เงื่อนไขก่อนทดสอบ: 1.สร้างอ็อบเจกต์ Bank เพื่อให้สามารถใช้ฟังก์ชันที่เกี่ยวข้องกับการจัดการบัญชีได้ ,</p> <p>2.สร้างอ็อบเจกต์ Account อย่างน้อยหนึ่งบัญชี เช่น account1 และ account2 เพื่อใช้ในการทดสอบ</p> <p>3.บัญชีที่สร้างขึ้นจะต้องมีข้อมูลที่ถูกต้อง เช่น ชื่อเจ้าของบัญชี เลขบัญชี และยอดเงินเริ่มต้น</p>						
<p>เงื่อนไขหลังทดสอบ: 1.จำนวนบัญชีในอ็อบเจกต์ Bank ต้องตรงตามที่คาดหวังหลังจากการดำเนินการทดสอบ</p> <p>2.ข้อมูลในบัญชีที่ทดสอบต้องยังคงสมบูรณ์และไม่ถูกเปลี่ยนแปลงหรือสูญหาย</p> <p>3.ยอดเงินในบัญชีหลังการทำธุรกรรม (ฝาก/ถอน) จะต้องตรงกับยอดเงินที่คาดหวัง</p> <p>4.อ็อบเจกต์ Bank และ Account ต้องถูกสร้างขึ้นใหม่ในการทดสอบแต่ละรายการเพื่อให้แน่ใจว่าไม่มีผลกระทบจากการทดสอบก่อนหน้า</p>						
<p>ขั้นตอนการดำเนินการทดสอบ: 1.สร้างอ็อบเจกต์ Bank และบัญชี account1, account2 ด้วยข้อมูลเริ่มต้น</p> <p>2.ดำเนินการทดสอบ Add Account , Remove Account, Deposit Money, Withdraw Money, Get Accounts</p> <p>3.ใช้ assertEquals และ assertTrue เพื่อตรวจสอบจำนวนบัญชีและยอดเงินในบัญชีตามที่คาดหวัง</p>						
รหัส	กรณีทดสอบ	ข้อมูลทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลลัพธ์ที่เกิดขึ้นจริง	ผลการทดสอบ	หมายเหตุ
T001	เพิ่มบัญชีใหม่เข้าในระบบธนาคาร addAccount() สร้างอ็อบเจกต์ Bank และ Account แล้ว เรียกฟังก์ชัน addAccount(account1) เพื่อ เพิ่มบัญชีเข้าสู่ระบบธนาคาร จากนั้น ตรวจสอบว่าจำนวนบัญชีในระบบเพิ่มขึ้น เป็น 1 บัญชี และใช้ assertEquals()เพื่อ ตรวจสอบจำนวนบัญชีและบัญชีที่เพิ่มเข้ามา	account1 : ชื่อ: "Habibah Masae" เลขบัญชี: "123456" ยอดเงิน: 500.0	จำนวนบัญชี: 1 บัญชีที่เพิ่ม: account1	จำนวนบัญชีในระบบ: 1 บัญชีที่เพิ่มเข้ามาคือ account1	ผ่าน	

T002	ลบบัญชีออกจากระบบธนาคาร removeAccount() สร้างอ็อบเจกต์ Bank และ Account เรียกฟังก์ชัน addAccount(account1) เพื่อ เพิ่มบัญชีเข้าสู่ระบบธนาคาร จากนั้นเรียก ฟังก์ชัน removeAccount(account1) เพื่อ ทำการลบบัญชี account1 และใช้ assertEquals()เพื่อตรวจสอบจำนวนบัญชี	account1 : ชื่อ: "Habibah Masae" เลขบัญชี: "123456" ยอดเงิน: 500.0	จำนวนบัญชีในธนาคาร: 0	จำนวนบัญชีในธนาคาร: 0	ผ่าน	
T003	ฝากเงินเข้าบัญชี depositMoney() สร้างอ็อบเจกต์ Bank และ Account เรียก ฟังก์ชัน addAccount(account1) เพื่อเพิ่ม บัญชีเข้าสู่ระบบธนาคาร และ เรียกฟังก์ชัน depositMoney(account1, 200.0) เพื่อ ฝากเงิน 200.0 บาทในบัญชี และใช้ assertEquals() เพื่อตรวจสอบยอดเงินในบัญชี	account1 : ชื่อ: "Habibah Masae" เลขบัญชี: "123456" ยอดเงิน: 500.0 depositMoney(account1) : 200.0	ยอดเงินในบัญชี account1: 700.0	ยอดเงินในบัญชี account1: 700.0	ผ่าน	
T004	ถอนเงินจากบัญชี withdrawMoney() สร้างอ็อบเจกต์ Bank และ account1และ เรียกใช้ฟังก์ชัน withdrawMoney(account1, 100.0) เพื่อ ถอนเงิน 100.0 บาท จากนั้น	account1 : ชื่อ: "Habibah Masae" เลขบัญชี: "123456" ยอดเงิน: 500.0 withdrawMoney(account1) : 100.0	ยอดเงินในบัญชี account1: 400.0	ยอดเงินในบัญชี account1: 400.0	ผ่าน	

T005	<p>ใช้assertEquals() เพื่อตรวจสอบยอดเงินในบัญชี</p> <p>ดึงข้อมูลบัญชี GetAccount() สร้างอ็อบเจกต์ Bank และ account1 และ account2 จากนั้นเรียกใช้ฟังก์ชัน addAccount(account1) และ addAccount(account2) จากนั้นใช้ assertEquals()เพื่อตรวจสอบจำนวนบัญชี และใช้ assertTrue() เพื่อตรวจสอบว่ามีบัญชีที่ต้องการอยู่ในระบบ</p>	<p>account1 : ชื่อ: "Habibah Masae", เลขบัญชี: "123456", ยอดเงิน: 500.0</p> <p>account2 : ชื่อ: "Onnicha Srisaman", เลขบัญชี: "654321", ยอดเงิน: 1000.0</p>	<p>จำนวนบัญชีในธนาคาร: 2 มีบัญชี account1 และ account2</p>	<p>จำนวนบัญชีในธนาคาร: 2 มีบัญชี account1 และ account2</p>	ผ่าน	
------	---	---	--	--	------	--