

## แบบฝึกปฏิบัติ ครั้งที่ 11

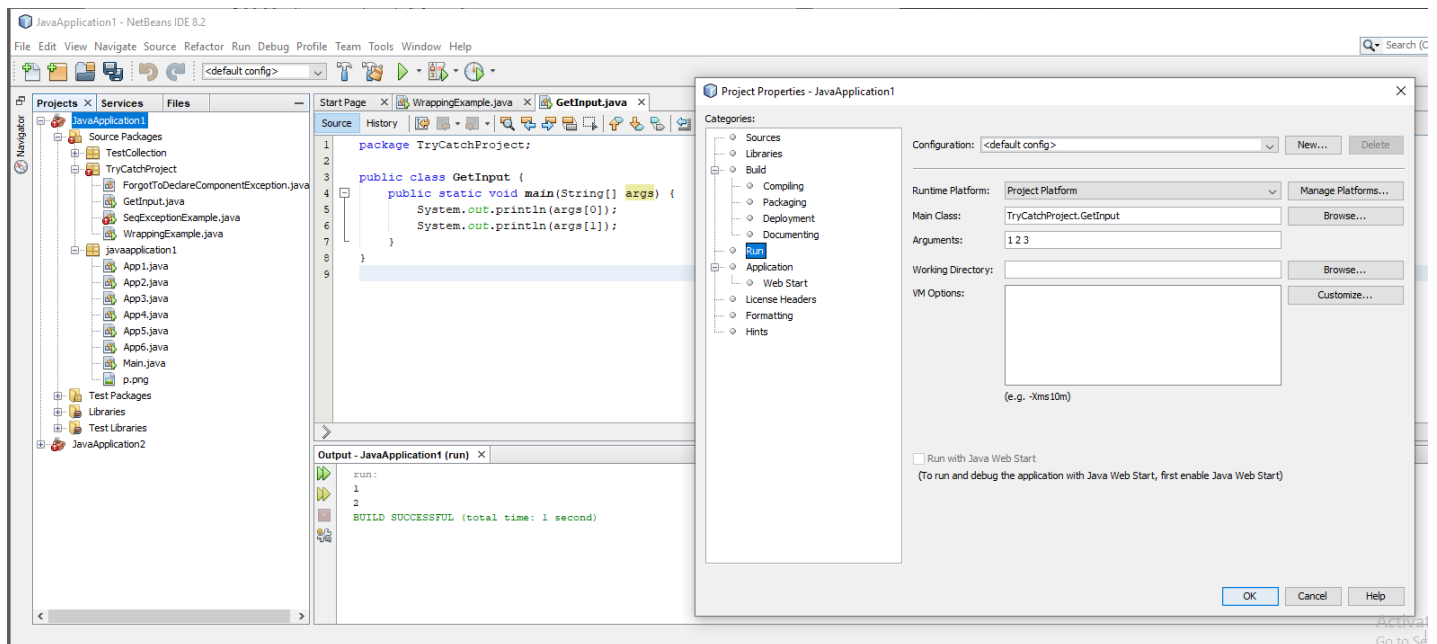
เรื่อง

การจัดการกับข้อผิดพลาด

วัตถุประสงค์


1. เพื่อให้เข้าใจข้อผิดพลาดที่อาจเกิดขึ้นได้ในภาษาจาวา
2. เพื่อให้เข้าใจคลาสที่เกี่ยวข้องกับการจัดการกับข้อผิดพลาดที่กำหนดไว้ในภาษาจาวา
3. เพื่อให้เข้าใจวิธีการเรียกใช้คำสั่งที่ใช้ในการจัดการกับข้อผิดพลาดคือ คำสั่ง try, catch และ finally
4. เพื่อให้สามารถจัดการกับข้อผิดพลาด และสร้างคลาสประเภทข้อผิดพลาดขึ้นมาใหม่ได้

### วิธีการรับค่าผ่าน Command Line ด้วยโปรแกรม Netbean



ขั้นที่ 1 คลิกขวาที่สัญลักษณ์ถ้วยกาแฟ ☕ . โปรเจค และเลือก Properties > Run จะปรากฏหน้าต่างย่อยดังภาพ

ขั้นที่ 2 เลือก Main Class เป็นไฟล์ที่ต้องการ Run จากนั้นกำหนดค่า input ในช่อง Arguments โดยที่แต่ละค่าจะถูกแบ่งด้วยช่องว่าง จากนั้นกด OK

ขั้นที่ 3 กด Run ที่ปุ่ม  เท่านั้น (ไม่สามารถคลิกขวารันไฟล์ได้)

## ข้อที่ 1 การจัดการกับข้อผิดพลาด

กำหนดให้ เมธอด `main()` ในเอกสาร หมายถึง เมธอด `public static void main(String[] args)`

### ขั้นตอนการปฏิบัติการ

- 1) จงเขียนคลาสที่ชื่อ `ExceptionDemo` โดยมีเมธอด `main()` อยู่ภายในคลาส และมีคำสั่งต่าง ๆ ดังนี้
  - a. ให้เขียนคำสั่งเพื่อรับข้อมูลที่เป็นเลขจำนวนจริง 3 ค่า (ค่าสัมประสิทธิ์  $a$ ,  $b$  และ  $c$ ) ผ่านทาง command line เพื่อกำหนดเป็นค่าสัมประสิทธิ์ของสมการ  $ax^2 + bx + c = 0$  และกำหนดให้ใช้เมธอด `Double.parseDouble()` ในการแปลงข้อมูลที่รับเข้ามาชนิด `String` ให้เป็นชนิด `double`
  - b. ให้เขียนคำสั่งกำหนดค่าเพื่อคำนวณหาคำตอบของ  $x_1$  และ  $x_2$  จากสูตร
 
$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{และ} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$
  - c. ให้เขียนคำสั่งเพื่อแสดงผลลัพธ์ของ  $x_1$  และ  $x_2$  ออกทางจอภาพ
- 2) คอมไพล์โปรแกรม `ExceptionDemo.java`
- 3) ทดลองรันโปรแกรมโดยป้อนสัมประสิทธิ์ที่มีค่าต่างๆ เพื่อทดสอบว่าโปรแกรมคำนวณค่า  $x_1$  และ  $x_2$  ได้ถูกต้อง
- 4) ปรับปรุงโปรแกรมในคลาส `ExceptionDemo` โดยกำหนดให้ใช้คำสั่ง `try/catch` เพื่อสามารถจัดการกับข้อผิดพลาดต่างๆ ได้แก่
  - a. กรณีที่ผู้ใช้ป้อนค่า  $a$ ,  $b$  และ  $c$  ที่ไม่ใช่ตัวเลข โปรแกรมจะตรวจจับข้อผิดพลาดและแสดงข้อความว่า "Please input data in number format only."
  - b. กรณีที่ผู้ใช้ป้อนค่า  $a$ ,  $b$  และ  $c$  ไม่ครบทุกตัว โปรแกรมจะตรวจจับข้อผิดพลาดและแสดงข้อความว่า "Please enter 3 numbers as a, b, and c respectively."
- 5) คอมไพล์และทดลองรันโปรแกรมใหม่

### ตัวอย่างการรันโปรแกรมผ่าน Command line

```

C:\Projects>javac ExceptionDemo.java

C:\Projects>java ExceptionDemo 4 8 3
x1: -0.5
x2: -1.5

C:\Projects>java ExceptionDemo 4.5 9.0 3.5
x1: -0.5285954792089683
x2: -1.4714045207910316

C:\Projects>java ExceptionDemo 4 8 Hello
Please input data in number format only.

C:\Projects>java ExceptionDemo 4 8
Please enter 3 numbers as a, b, and c respectively.

C:\Projects>
  
```

## ข้อที่ 2 การสร้างคลาสประเภท Exception

### ขั้นตอนการปฏิบัติการ

- 1) คัดลอกไฟล์ Customer.java Account.java และ CheckingAccount.java จากบทที่ 5
- 2) จงเขียนคลาสที่ชื่อ WithdrawException โดยมีขั้นตอนดังนี้
  - a. ประกาศคลาสให้สืบทอดมาจากคลาส Exception
  - b. กำหนด constructor แบบ default โดยมีคำสั่ง super() อยู่ภายใน
  - c. กำหนด constructor ที่มี argument ที่ชื่อ s ซึ่งเป็น String และคำสั่ง super(s) อยู่ภายใน
- 3) จงปรับปรุงเมธอด withdraw() ที่อยู่ในคลาส Account และ CheckingAccount ให้มีรูปแบบดังนี้

```
public void withdraw(double amount) throws WithdrawException {  
  
}
```

- a. ให้ปรับปรุงคำสั่ง if ภายในเมธอด withdraw() ในกรณีที่ที่มีจำนวนเงินในบัญชีน้อยกว่ายอดเงินที่ต้องการจะถอน ให้ส่งอีอบเจกต์ของคลาส WithdrawException คืนมา ด้วยคำสั่ง  

```
throw new WithdrawException(String)
```
- b. กำหนด String ที่ส่งให้ constructor WithdrawException(String) เป็นข้อความ  
"Account <ชื่อ Account> has not enough money."

- 4) คอมไพล์และทดลองรันโปรแกรม Main.java ซึ่งมีโค้ดโปรแกรม ดังนี้

```
public static void main(String[] args) {  
  
    CheckingAccount acct1 = new CheckingAccount(1000, "A0001", 500);  
    Customer cust = new Customer("Sompong", "Sookjai", acct1);  
    cust.getAcct().deposit(500);  
    cust.getAcct().withdraw(1800);  
    cust.getAcct().withdraw(300);  
    cust.getAcct().deposit(1000);  
    cust.getAcct().withdraw(200);  
  
}
```

จากโปรแกรมข้างต้น โปรแกรมสามารถคอมไพล์ผ่านหรือไม่ กรณีที่คอมไพล์ไม่ผ่าน เป็นเพราะเหตุใด

ไม่ได้ เพราะว่าจะเกิด **WithdrawException** เกิดขึ้นตรง **withdraw(300)**  
เนื่องจากไม่ได้ใส่ **try-catch** เพื่อดักจับ **Exception**

- 5) ปรับปรุงโปรแกรม Main.java ข้างต้นให้สามารถจัดการกับข้อผิดพลาดที่เกิดขึ้น อีกทั้งมีการแสดงข้อความ "Thank you." เมื่อทำโปรแกรมเสร็จสิ้นทุกครั้ง (ไม่ว่าจะมี Exception เกิดขึ้นหรือไม่ก็ตาม) โดยกำหนดให้ใช้คำสั่ง try/catch และ finally ซึ่งเมื่อรันแล้ว มีผลลัพธ์ดังนี้

```
500.0 baht is deposited to A0001.  
1800.0 baht is withdrawn from A0001 and your credit balance is 200.0.  
WithdrawException: Account A0001 has not enough money!  
Thank you.
```