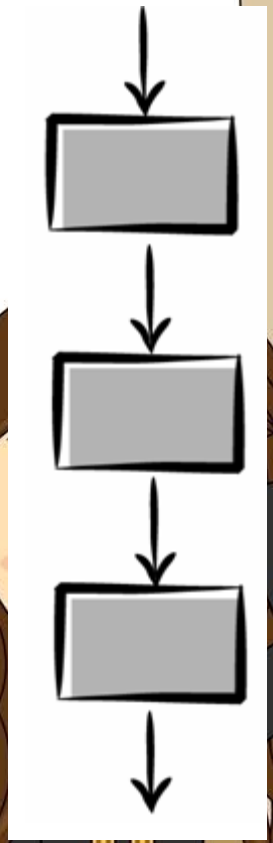
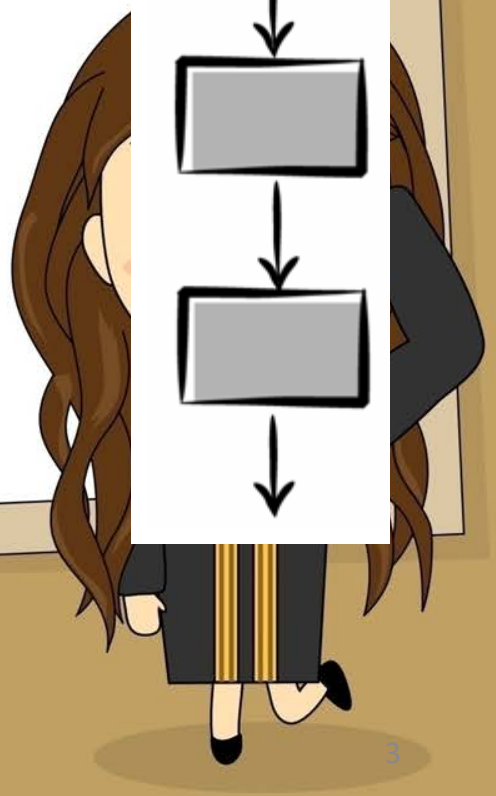
ประโยคควบคุม

หัวข้อ

- การทำงานแบบเรียงลำดับ
- ประโยค if, if-else
- ประโยค switch
- ประโยคที่ใช้ทำงานซ้ำหรือลูป
  - for
  - while
  - do-while
- คำสั่ง break
- คำสั่ง continue

การทำงานแบบเรียงลำดับ

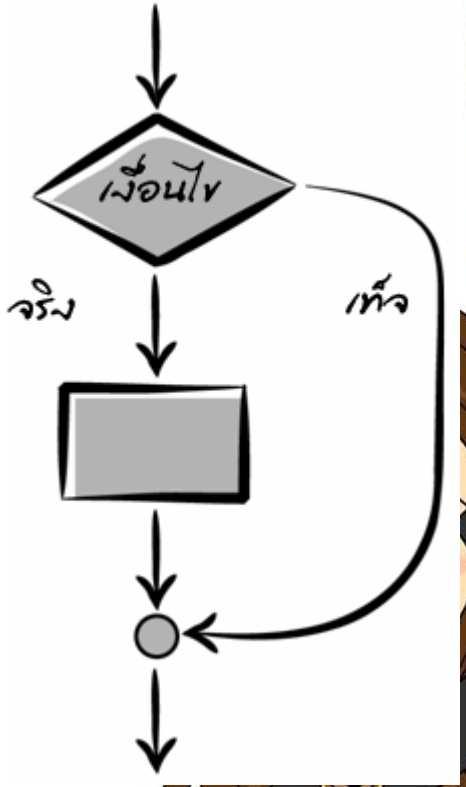


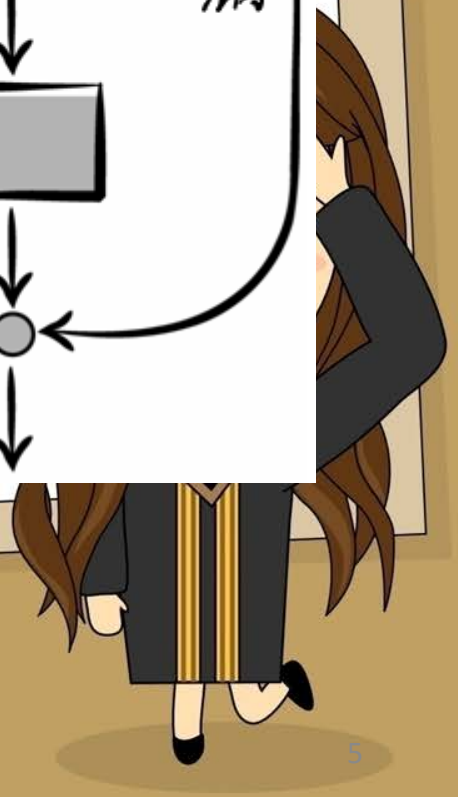


การทำงานแบบเรียงลำดับ

```
public class Shopping
{
   public static void main(String[] args)
   {
      int cash      = 500;
      int bookPrice = 180;
      int foodPrice =  20;
      int sodaPrice =   7;
      cash -= bookPrice;
      cash -= foodPrice;
      cash -= sodaPrice;
      System.out.println("Cash = " + cash);
   }
}
```

ประโยค if





ประโยค if

- รูปแบบ

```
if (เงื่อนไข)
{
    ประโยคที่จะทำงานเมื่อเงื่อนไขเป็นจริง;
}
```

- ตัวอย่าง

```
if (cash >= bookPrice)  {
   System.out.println("Buy book.");
}
```

ตัวอย่างการตัดสินใจซื้อหนังสือ

```
public class If {
    public static void main(String[] args)    {
        int cash      = 200;
        int bookPrice = 180;

        if (cash >= bookPrice)  {
            cash -= bookPrice;
            System.out.println("Buy book.");
        }

        System.out.println("Cash = " + cash);
    }
}
```

การเปรียบเทียบ

1. <
2. >
3. <=
4. >=
5. ==
6. !=

การเปรียบเทียบทศนิยม

- แทนที่จะเทียบความเท่ากัน บางทีควรเทียบว่ามีค่าใกล้เคียงกันหรือไม่
- เช่น d = √2
  - d * d ไม่เท่ากับ2
  - d * d มีค่าใกล้เคียงกับ 2

ตัวดำเนินการแบบบูล (Boolean operator)

- && คือ AND
  - เช่น เงื่อนไข cash >= price && price < 150
- || คือ OR
- ! คือ NOT

ตัวดำเนินการแบบบูล

เงื่อนไข		ตัวดำเนินการแบบบูล		
A	B	A && B	A || B	!A
false	false	false	false	true
false	true	false	true	true
true	false	false	true	false
true	true	true	true	false

ตัวดำเนินการแบบบูล

```
public class IfAnd
{
   public static void main(String[] args)
   {
      int cash  = 220;
      int price = 100;

      if (cash >= price && price < 150)
         System.out.println("Buy book.");
   }
}
```

การใช้ตัวแปรแทนเงื่อนไข

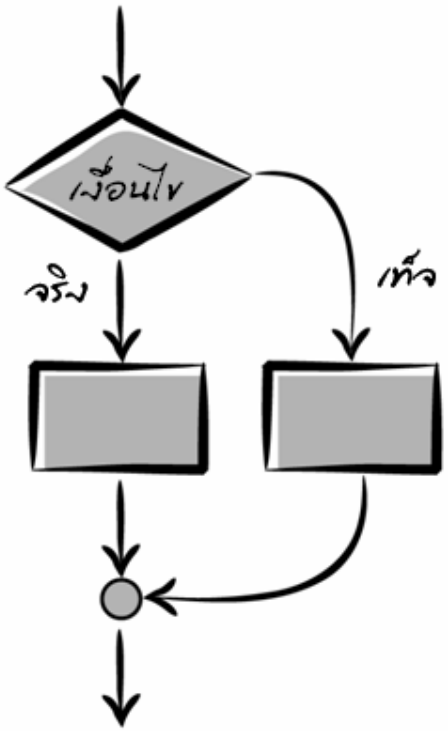
- if (มีเงินพอ && ราคาที่สมเหตุผล)

- if (cash >= price && price < 150)

- boolean hasEnoughCash = cash >= price;
  boolean reasonablePrice = price < 150;
  if (hasEnoughCash && reasonablePrice)

การทำงานเมื่อเงื่อนไขเป็นเท็จ (else)

- รูปแบบ

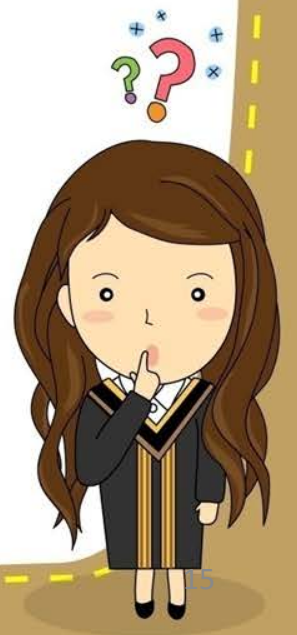



```
if (เงื่อนไข)
{
   ประโยคที่จะทำงานเมื่อเงื่อนไขเป็นจริง;
}
else
{
   ประโยคที่จะทำงานเมื่อเงื่อนไขเป็นเท็จ;
}
```

ตัวอย่างการตัดสินใจใช้บัตรเครดิต

```
 int cash  = 200;
int price = 380;

if (cash >= price)
{
   cash -= price;
   System.out.println("Use cash");
}
else
{
   System.out.println("Use card");
}
system.out.println(“Cash = “ + cash);
}
}
```



การกำหนดค่าตามเงื่อนไข

- รูปแบบ
  - *ตัวแปร* = *เงื่อนไข* ? *a* : *b*;
- ตัวอย่าง

```
int price = 99;
int withdraw = price <= 100 ? 100 : 200;
```

ถ้าเงื่อนไขเป็นจริง withdraw มีค่าเป็น 100 ถ้าเงื่อนไขเป็นเท็จ withdraw มีค่าเป็น 200

การจัดระเบียบโปรแกรม

```
public class IfAssignMessy
{
public static void main( String[] args )
{
int price = 99;
int withdraw;
if( price <= 100 )
withdraw = 100;
else
withdraw = 200;
System.out.println("Withdraw = " + withdraw);
}
}
```

การจัดระเบียบโปรแกรม

```
public class IfAssign
{
   public static void main(String[] args)
   {
      int price = 99;
      int withdraw;

      if (price <= 100)
         withdraw = 100;
      else
         withdraw = 200;

      System.out.println("Withdraw = " + withdraw);
   }
}
```

การจัดระเบียบโปรแกรม

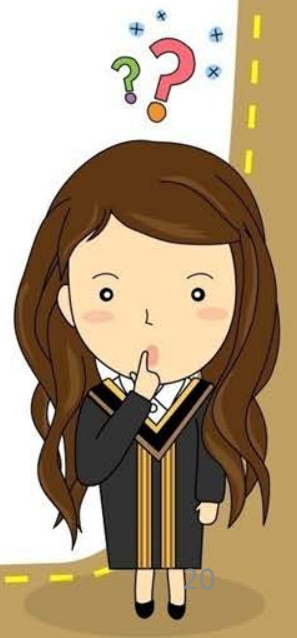
```
public class IfAssign {
    public static void main(String[] args) {
        int price = 99;
        int withdraw;

        if (price <= 100)
            withdraw = 100;
        else
            withdraw = 200;

        System.out.println("Withdraw = " + withdraw);
    }
}
```

รูปแบบ if-else if

```
if (เงื่อนไข1)
{
    ประโยคที่จะทำงานเมื่อ เงื่อนไข1 เป็น จริง;
}
else if (เงื่อนไข2)
{
    ประโยคที่จะทำงานเมื่อ เงื่อนไข2 เป็น จริง;
}
else
{
    ประโยคที่จะทำงานเมื่อ เงื่อนไขก่อนหน้านี้ทั้งหมด เป็น เท็จ;
}
```

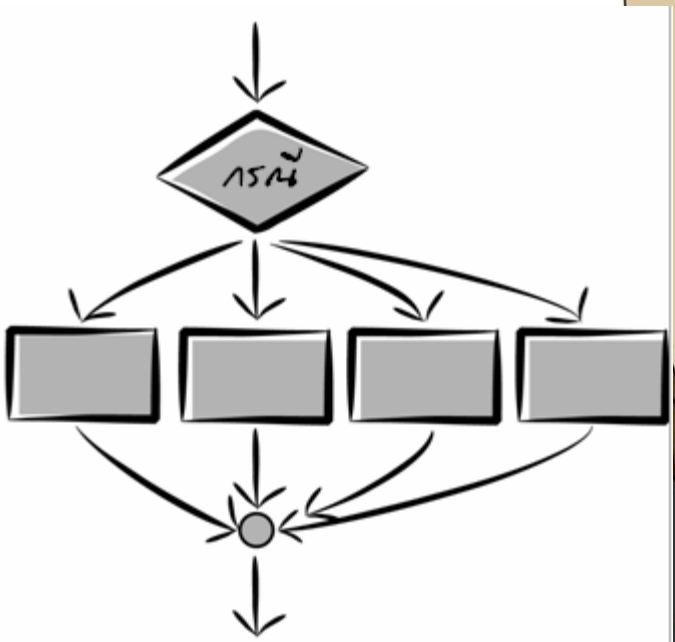


ตัวอย่างการคิดเกรด

```
char grade;
int score = 70;

if (score >= 90)
   grade = 'A';
else if (score >= 80)
   grade = 'B';
else if (score >= 60)
   grade = 'C';
else if (score >= 40)
   grade = 'D';
else
   grade = 'F';
```

ประโยค swit

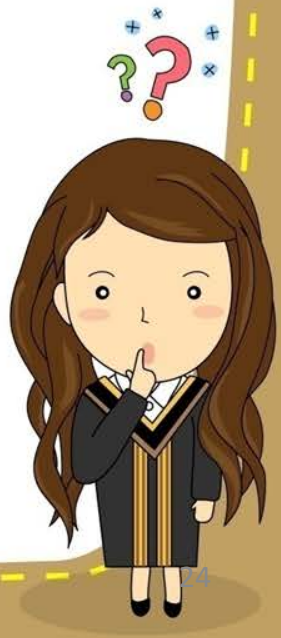


รูปแบบ

```
switch (กรณี)
{
case กรณี1:
   ประโยค;
   break;
case กรณี2:
   ประโยค;
   break;
default:
   ประโยค;
}
```

ตัวอย่างการพิมพ์ข้อความชมเชยตามเกรดที่ได้

```
char grade = 'B';
switch (grade)
{
case 'A':
   System.out.println(“ดีมาก");
   break;
case 'B':
   System.out.println(“ดี");
   break;
case 'C':
   System.out.println(“พอใช้");
   break;
case 'D':
   System.out.println(“ต้องปรับปรุง");
   break;
default:
   System.out.println(“แก้ไข");
}
```



Menu

1. บวก
2. ลบ
3. คูณ
4. หาร

แบบฝึกหัด

ให้รับ ชื่อผู้ซื้อ และเลือกหนังสือ และให้กำหนดจำนวนหนังสือ โดยมีเมนูดังนี้

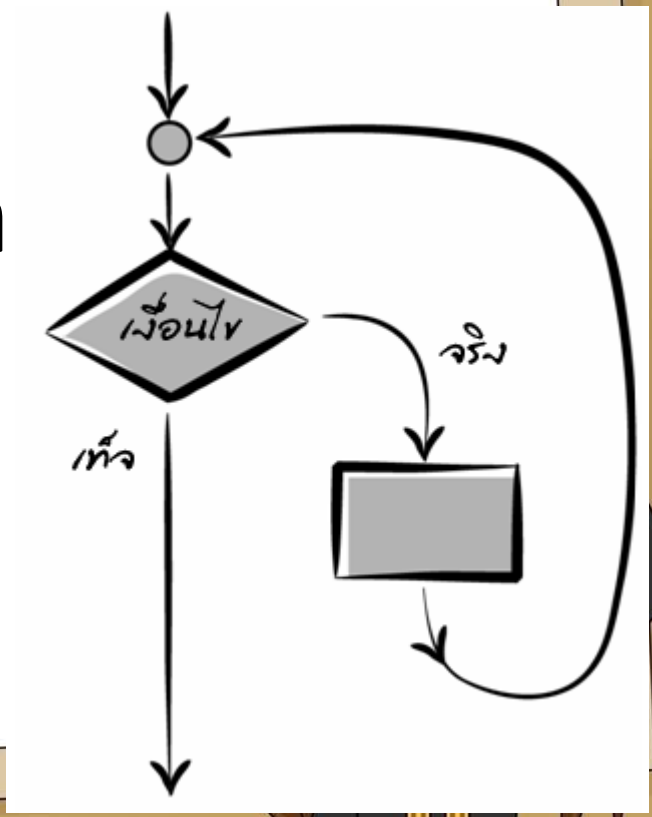
MENU

1. การเขียนโปรแกรมภาษาจาวา
2. คู่มือ VB.NET
3. หนังสือภาษาไทย

โดยกำหนดว่า หนังสือการเขียนโปรแกรมภาษาจาวา ราคา 350

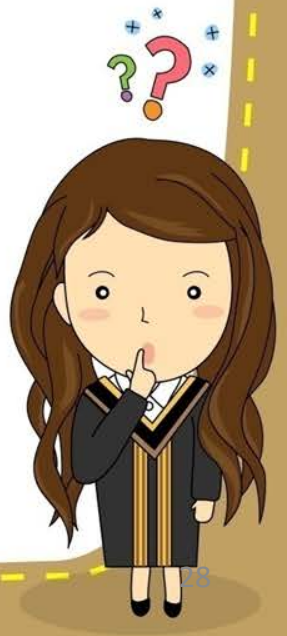
คู่มือ VB.NET 540 หนังสือภาษาไทย 150

การวนลูปแบบ wh



รูปแบบ

```
while (เงื่อนไข)
{
    ประโยคที่ทำซ้ำขณะที่เงื่อนไขเป็นจริง;
}
```

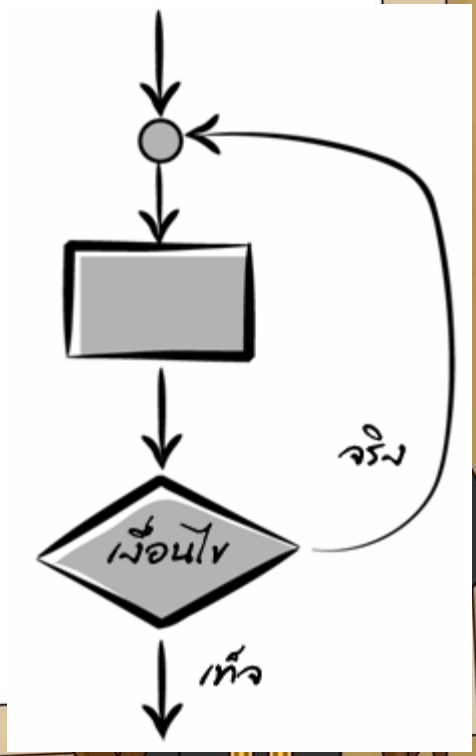


ตัวอย่างการคำนวณดอกเบี้ย

```
int year = 0;

double balance = 100;
while (balance <= 10000)
{
    year++;
    balance *= 1.05;
}
```

การวนลูปแบบ do-while



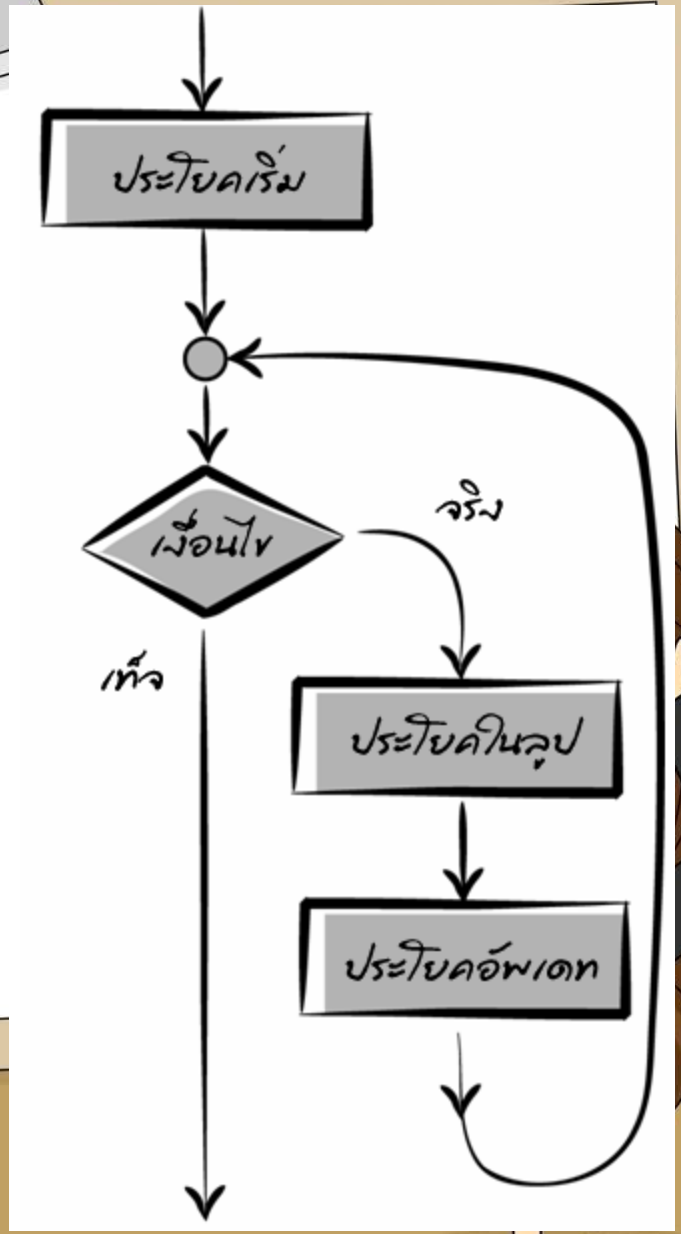
รูปแบบ

```
do
{
    ประโยค;
} while (เงื่อนไข);
```

ตัวอย่างวนลูปจนกว่าจะได้ข้อมูลที่ถูกต้อง

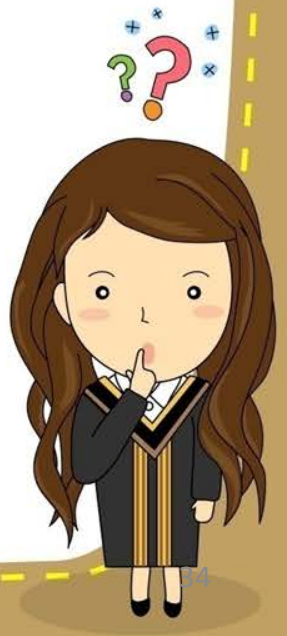
```
do
{
    รับข้อมูล;
} while (ข้อมูลไม่ถูกต้อง);
นำข้อมูลที่ถูกต้องไปประมวลผล;
```

การวนลูปแบบ for



รูปแบบ

```
for (ประโยคเริ่ม; เงื่อนไข; ประโยคอัพเดท)
{
    ประโยคในลูป;
}
```



ตัวอย่างวนลูปเป็นจำนวน 5 ครั้ง

```
for (int i = 0; i < 5; i++)
{
    // ประโยคที่จะทำซ้ำ 5 ครั้ง
}
```

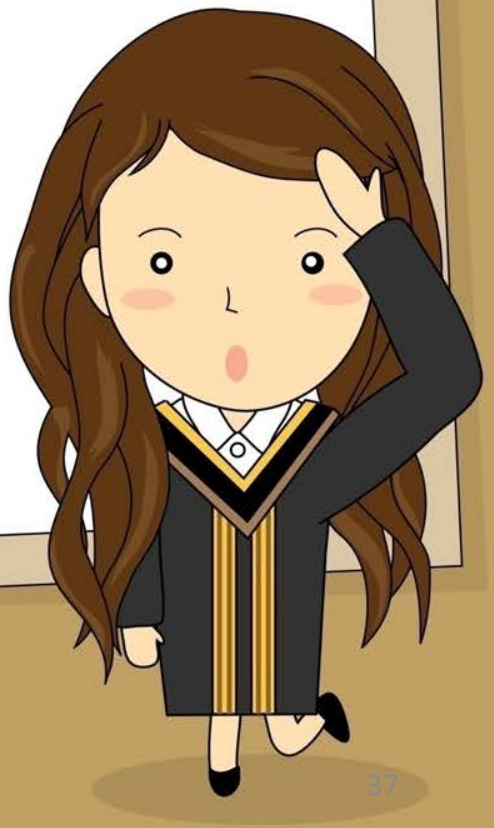
ตัวอย่างการคำนวณยอดเงินในบัญชีเมื่อเวลาผ่านไป 95 ปี

```
double balance = 100;


for (int i = 0; i < 95; i++)
{
    balance *= 1.05;  (balance = balance*1.05)
}

System.out.println(balance);
```

คำสั่ง break



การออกจาก while ด้วยคำสั่ง break

```
while( เงื่อนไข ในการวนลูป)
{
    ....
    if( เงื่อนไขที่จะออกจากลูป )
        break;
    ....
}

คำสั่งหลังลูป;
```

การออกจาก for ด้วยคำสั่ง break

```
for (int i = 0; i < 5; i++)
{
    System.out.print("<");

    if (i == 2)
        break;

    System.out.print(i + ">");
}
```

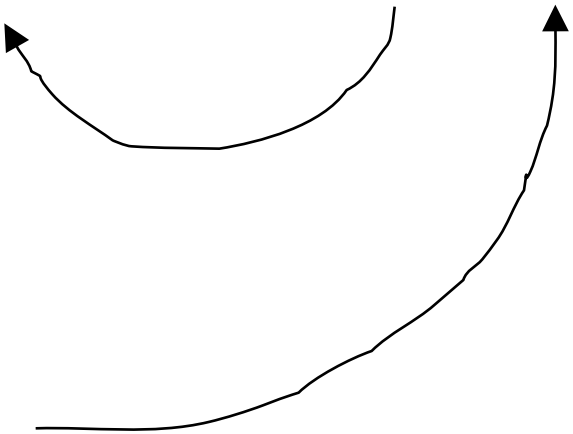
คำสั่ง continue

ข้ามไปตรวจสอบเงื่อนไข

```
while( เงื่อนไขในการวนลูป)
{
        ....
        if( เงื่อนไขที่จะกลับไปต้นลูป )
                continue;
        ....
}
```

ข้ามไปตรวจสอบเงื่อนไข

```
for (ประโยคเริ่ม; เงื่อนไข; ประโยคอัพเดท)
{
        ....
        if( เงื่อนไขที่จะกลับไปต้นลูป )
                continue;
        ....
}
```



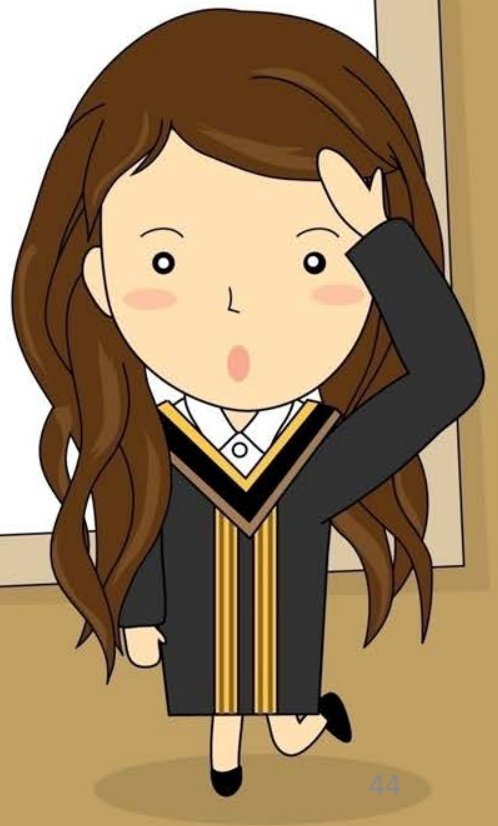
ตัวอย่างโปรแกรม

```
for (int i = 0; i < 5 ;  i++ )
{
   System.out.print("<");

   if (i == 2)
      continue;

   System.out.print(i + ">");
}
```

สรุป



สรุป

- โดยทั่วไปโปรแกรมจะทำงานเรียงลำดับทีละคำสั่ง
- ประโยคควบคุมมีหน้าที่ควบคุมการลำดับการทำงานของคำสั่งต่างๆ
- คำสั่งที่อยู่ถัดจากประโยค if จะทำงานเมื่อเงื่อนไขเป็นจริง

สรุป

- ประโยค **if-else** ทำให้โปรแกรมทำงานอย่างหนึ่งเมื่อเงื่อนไขเป็นจริง และทำงานอีกอย่างหนึ่งเมื่อเงื่อนไขเป็นเท็จ
- ประโยค **if-else if** ใช้กรณีที่มีหลายๆ เงื่อนไข
- ประโยค **switch** ใช้กรณีที่มีหลายๆ เงื่อนไข โดยจะเปรียบเทียบกับค่าคงที่

สรุป

- ประโยค while ใช้เพื่อวนทำงานซ้ำจนกว่าเงื่อนไขจะเป็นเท็จ
- ประโยค do-while จะทำงานในกลุ่มประโยคหลัง do หนึ่งครั้ง ตรวจสอบเงื่อนไข และจะวนลูปไปเรื่อยๆจนกว่าเงื่อนไขจะเป็นเท็จ
- ประโยค for เป็นลูปที่เหมาะสำหรับการกำหนดจำนวนครั้งการวนซ้ำ

สรุป

- คำสั่ง break ใช้เพื่อออกจากลูป
- คำสั่ง continue จะตรวจสอบเงื่อนไขของลูป ถ้าเป็นจริงจะทำงานคำสั่งที่อยู่ต้นลูป แต่ถ้าเป็นเท็จจะออกจากลูป

แบบฝึกหัด

- จงเขียนโปรแกรมเพื่อแสดงชื่อและเงินเดือนโดยใช้คำสั่ง if- else ที่ประมวลผลแสดงผลดังต่อไปนี้

- เงินเดือนมากกว่า 200000 แสดงผล คุณเป็นผู้จัดการ
- เงินเดือนมากว่า 150000 – 200000 แสดงผลคุณเป็นผู้จัดการฝ่าย
- เงินเดือนมากกว่า 100000 – 150000 แสดงผลคุณเป็นผู้จัดการแผนก
- เงินเดือนตั้งแต่ 50000 – 100000 แสดงผลคุณเป็นหัวหน้าหน่วย
- เงินเดือนต่ำกว่า 50000 แสดงผลคุณเป็นพนักงาน

แบบทดสอบ

ให้เขียนโปรแกรมตรวจสอบ **Username = add** และ

Password = add

จำนวน 3 ครั้ง ***ถ้าถูกครั้งใดครั้งหนึ่งแสดงว่า **ATC**

ครั้งที่ 1 แสดงข้อความว่า กรุณาตรวจสอบครั้งที่ 2

ครั้งที่ 2 แสดงข้อความว่า กรุณาตรวจสอบครั้งที่ 3

ครั้งที่ 3 แสดงข้อความว่า ออกจากโปรแกรม