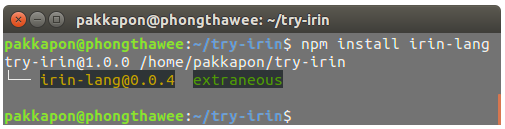
**คู่มือการติดตั้งอย่างละเอียด**

**การดาวน์โหลด**

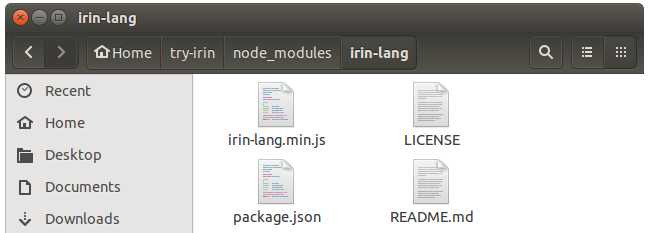
ท่านสามารถติดตั้งตัวแปลภาษาไอรินเพื่อใช้กับโปรแกรมของท่านได้โดยการติดตั้งได้ผ่านทาง npm,bower หรือสามารถติดตั้งผ่านการดาวน์โหลดไฟล์ที่ถูกแปลงเป็นจาวาสคริปและลดขนาดแล้วได้

**การติดตั้งผ่าน npm**

หากท่านใช้งาน npm ท่านสามารถติดตั้งตัวแปลภาษาไอรินได้โดยการใช้งานคำสั่ง npm install irin-lang



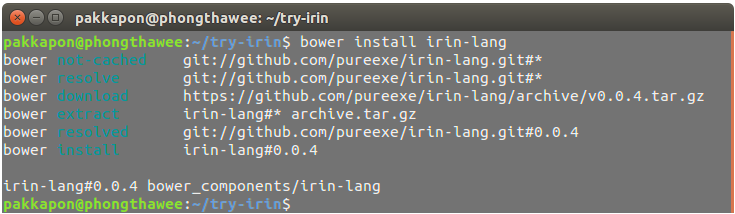
เมื่อการติดตั้งเสร็จสมบูรณ์จะปรากฏข้อความดังภาพ



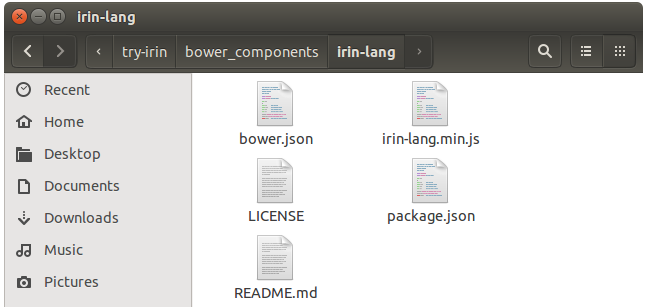
จะปรากฏไฟล์ขึ้นมาที่ โฟลเดอร์/node\_modules/irin-lang

**การติดตั้งผ่าน bower**

หากท่านใช้งาน bower ท่านสามารถติดตั้งตัวแปลภาษาไอรินได้โดยการใช้งานคำสั่ง bower install irin-lang



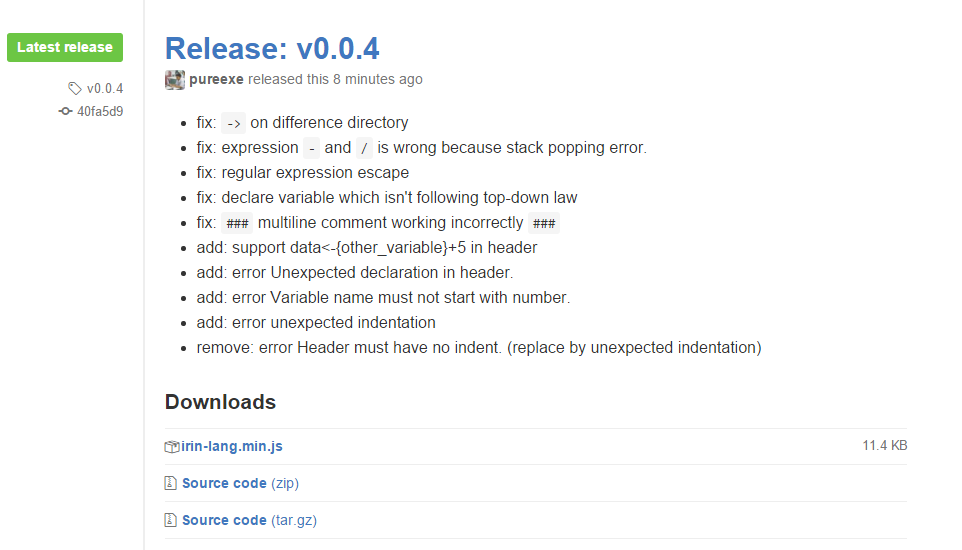
เมื่อการติดตั้งเสร็จสมบูรณ์จะปรากฏข้อความดังภาพ



จะปรากฏไฟล์ขึ้นที่ โฟลเดอร์/bower\_components/irin-lang

**การติดตั้งโดยการดาวน์โหลดไฟล์**

ท่านสามารถดาวน์โหลดไฟล์ได้ที่ https://github.com/pureexe/irin-lang/releases



โดยให้ท่านดาน์โหลดไฟล์ชื่อ irin-lang.min.js เพื่อนำไปใช้งาน

**การติดตั้ง**

**ใช้งานผ่าน Node.JS หรือ CommonJS**

ในการใช้งานผ่าน Node.JS หรือ CommonJS สามารถนำเข้าตัวแปลภาษาได้ผ่านทางคำสั่ง require โดยสามารถเขียนคำสั่งได้ดังนี้

var Irin = require(“irin-lang”)

**ใช้งานผ่านเบราว์เซอร์**

ในการใช้งานผ่านเบราวเซอร์ สามารถนำเข้าตัวแปลภาษาได้ผ่านทางแท็ก script โดยสามารถเขียนคำสั่งได้ดังนี้

<script src=”path/to/irin-lang.min.js”></script>

**การใช้งานในโปรแกรม**

**การแปลภาษา**

ก่อนที่จะเริ่มโต้ตอบกับระบบสนทนาโต้ตอบอัตโนมัติที่เขียนขึ้นจากภาษาไอรินได้ จำเป็นต้องทำการแปลภาษาก่อน ซึ่งสามารถทำการแปลภาษาได้โดยการเขียนโค้ดภาษาจาวาสคริปดังนี้

var bot = new Irin(“ที่อยู่ไฟล์.irin”,function(err){

if(err){

//เกิดข้อผิดพลาดขึ้นขณะแปลภาษา

}else{

//การแปลภาษาเสร็จสมบูรณ์

}

});

**การโต้ตอบ**

การโต้ตอบกับโค้ดภาษาไอรินที่ถูกแปลภาษาเรียบร้อยแล้วนั้นทำได้โดยการเขียนโค้ดภาษาจาวาสคริปดังนี้

bot.reply(“คำถาม”)

**การใช้งานภาษาไอริน**

การเยื้องเพื่อการรู้จำภาษาธรรมชาติอย่างชาญฉลาด (Indent to Recognize for Intelligent Natural language : IRIN ) หรือเรียกอีกชื่อว่า ภาษาไอริน เป็นภาษาที่แบบมาให้ใช้การเยื้องในการแทนตำแหน่งของคำถามและคำตอบในการสร้างระบบสนทนาโต้ตอบอัตโนมัติซึ่งจะทำให้โค้ดอ่านง่ายและสั้นลงอย่างมาก ซึ่งมีไวยากรณ์ในการเขียนดังนี้

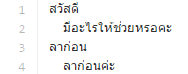
**การเยื้อง**

ในภาษาไอรินใช้การเยื้องในการแทนตำแหน่งของคำถามและคำตอบโดยบรรทัดที่ไม่มีการเยื้องเลยจะทำหน้าที่เป็นคำถาม และบรรทัดที่เยื้องเข้าไปจะเป็นคำตอบ เมื่อเยื้องเข้าไปอีกจะเป็นคำถาม สลับกันไปเรื่อยๆ ตัวอย่างเช่น



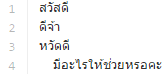
เมื่อถามระบบว่า สวัสดี ระบบจะตอบว่า มีอะไรให้ช่วยหรอคะ ซึ่งจะเห็นได้ว่า สวัสดี ทำหน้าที่เป็นคำถาม และ มีอะไรให้ช่วยหรอค่ะ เป็นคำตอบ

อีกทั้งภาษาไอรินยังใช้การเยื้องในการแยกคำถามแต่ละข้อออกจากกัน ตัวอย่างเช่น



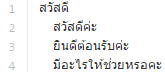
จะเห็นว่าโค้ดดังกล่าวมี 2 คำถามคือ สวัสดี กับลาก่อน

แต่หากไม่มีการเยื้องในบรรทัดที่ติดกันจะถือว่าเป็นคำถามที่มีคำตอบร่วมกัน ตัวอย่างเช่น



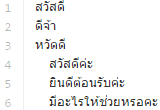
จะสังเกตว่าบรรทัดที่ 1,2 และ 3 ไม่มีการเยื้อง จึงถือว่าเป็นคำถามร่วมกัน ดังนั้นเมื่อถามระบบว่า สวัสดี, ดีจ้า, หรือหวัดดี ระบบจะตอบกลับมาเหมือนกันคือ มีอะไรให้ช่วยหรอคะ

และหากไม่มีการเยื้องในบรรทัดที่ติดกันของคำตอบจะถือว่าคำตอบนั้นเป็นคำตอบร่วมกันของคำถามโดยคำตอบร่วมกัน จะใช้การสุ่มหนึ่งในคำตอบออกมาตอบ ตัวอย่างเช่น



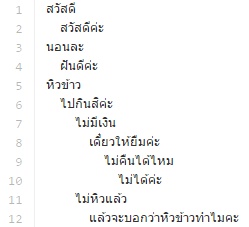
หากเราถามระบบว่าสวัสดี ระบบอาจตอบว่า สวัสดีค่ะ, ยินดีต้อนรับค่ะ หรือมีอะไรให้ช่วยหรอคะ

ทำนองเดียวกันเราสามารถใช้คำถามร่วมและคำตอบร่วมพร้อมกันได้ ตัวอย่างเช่น



เมื่อถามระบบว่าสวัสดี,ดีจ้า หรือหวัดดี ระบบอาจตอบว่า สวัสดีค่ะ, ยินดีต้อนรับค่ะ หรือมีอะไรให้ช่วยหรอคะ

อีกทั้งเรายังสามารถใช้การเยื้องเพื่อเป็นการบอกการสนทนาต่อเนื่องได้ โดยการเพิ่มการเยื้องเข้าไปต่อจากคำตอบของคำถามก่อนหน้า ตัวอย่างเช่น



สังเกตว่าบริเวณคำว่าหิวข้าวมีการเยื้องถัดเข้าไปต่อจากคำตอบของคำถามแรก นั่นหมายถึงการสนทนาต่อเนื่อง เมื่อถามระบบว่า สวัสดี ตัวแปลภาษาจะตอบว่า สวัสดีค่ะ แต่เมื่อเราถามระบบต่อด้วยคำว่า หิวข้าว ระบบจะตอบกลับมาว่า ไปกินสิค่ะ ซึ่งช่วงนี้จะเป็นการสนทนาต่อเนื่องซึ่งตอนนี้มี 2 ทางเลือกคือ ไม่มีเงิน กับไม่หิวแล้ว ซึ่งหากถามระบบต่อว่าว่าไม่มีเงิน ระบบจะตอบว่า เดี๋ยวให้ยืมค่ะ แต่หากเรากรอกว่านอนละ ระบบจะออกจากการสนทนาต่อเนื่องแล้วตอบว่า ฝันดีค่ะ

**ความคิดเห็น**

ความคิดเห็นหรือคอมเมนต์ (Comment) ใช้เพื่อบอกความคิดเห็นของเราลงไปในโค้ด โดยความคิดเห็นนั้นมีไว้เพื่อให้มนุษย์อ่านเท่านั้น เพื่อให้ผู้เขียนโค้ดสามารถจำได้ว่าโค้ดที่ตัวเองเขียนมีไว้ใช้งานอย่างไร โดยความคิดเห็นนั้นจะไม่ถูกนำไปประมวลผลโดยตัวแปลภาษา

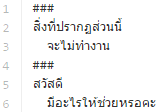
**ความคิดเห็นบรรทัดเดียว**

ใช้เครื่องหมาย # เพื่อระบุว่าเป็นความคิดเห็นบรรทัดเดียว โดยตัวแปลภาษาจะไม่นำข้อความหลังเครื่องหมายไปประมวลผล ตัวอย่างเช่น



**ความคิดเห็นหลายบรรทัด**

ใช้เครื่องหมาย ### เพื่อระบุว่าเป็นความคิดเห็นหลายบรรทัด โดยต้องปิดท้ายความเห็น ด้วย ### เสมอ โดยตัวแปลภาษาจะไม่นำข้อความที่อยู่ในช่วงของเครื่องหมายดังกล่าวไปประมวลผล ตัวอย่างเช่น

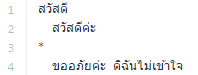


**นิพจน์**

นิพจน์ (Expression) คือ รูปแบบสำหรับเปรียบเทียบข้อความว่าเข้าเงื่อนไขหรือไม่ หานิพจน์ตรงกับรูปแบบของศัพท์ที่กรอกเข้ามาจะทำการเรียกคำตอบของคำถามนั้นถัดไป โดยภาษาไอรินมีนิพจน์ต่างๆ ดังต่อไปนี้

**ดอกจันทร์**

ดอกจันทร์ \* คือ ตรงกับทุกตัวอักษร ไม่ว่าตัวอักษรใดก็ตามจะเป็นจริงเสมอ ตัวอย่างเช่น



เมื่อเราถามคำถามใดกับระบบก็ตามที่ไม่ใช่ สวัสดี ระบบจะตอบกลับมาว่า ขออภัยค่ะ ดิฉันไม่เข้าใจ เนื่องจากนิพจน์ดอกจันทร์แทนตัวอักษรใดก็ได้

นอกจากนี้เรายังใช้นิพจน์ดอกจันทร์แทนบางส่วนของประโยคได้เช่น



เมื่อถามระบบว่า ผมว่าเธอน่ารักมาก กับถามระบบว่า ผมว่าเธอสวยมาก ระบบจะตอบกลับมาว่า ทำไมคุณถึงคิดอย่างงั้นละ

แต่ว่าดอกจันทร์นั้นไม่ถือว่าตรงกับความว่างเปล่า ต้องมีอย่างน้อย 1 ตัวอักษรขึ้นไป ตัวอย่างเช่นโค้ดด้านบน หากถามระบบว่า ผมว่าเธอมาก จะสังเกตได้ว่า บริเวณที่เป็นดอกจันทร์จะไม่ตรงกับตัวอักษรใดๆเลย ดังนั้นระบบจะไม่ตอบว่าทำไมคุณถึงคิดอย่างงั้นละ จึงขอให้ระวังในส่วนนี้เอาไว้ด้วย

**วงเล็บ**

( ) คือ เลือก คำใดคำหนึ่งจากในวงเล็บ แบ่งคำออกจากกันด้วยเครื่องหมาย | ตัวอย่างเช่น



เมื่อถามระบบว่า หิวไหม หรือถามว่า หิวหรือไม่ ระบบจะตอบกลับมาว่า หิวแล้วแล้วค่ะ เนื่องจากคำว่าไหมและคำว่าหรือไม่เป็นคำใดคำหนึ่งจากในวงเล็บ แต่หากถามว่าหิวมะ ระบบจะไม่ตอบว่าหิวแล้วค่ะ เนื่องจากคำว่ามะ ไม่ได้ตรงกับคำใดคำหนึ่งในวงเล็บ

**วงเล็บก้ามปู**

[ ] คือ อาจ มีคำใดคำหนึ่งจากในวงเล็บ แบ่งคำออกจากกันด้วยเครื่องหมาย | ตัวอย่างเช่น



เมื่อถามระบบว่า สวัสดี,สวัสดีครับ หรือสวัสดีค่ะ ระบบจะตอบกลับมาว่า สวัสดีค่ะ แต่หากถามระบบว่าสวัสดีนะ ระบบจะไม่ตอบกลับมาว่าสวัสดีค่ะ เนื่อจากไม่ได้บอกกับระบบว่าจากมีคำว่า นะ อยู่ด้านหลัง

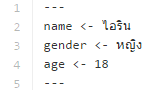
และเรายังสามารถใช้เครื่องหมายวงเล็บก้ามปูร่วมกับดอกจันทร์ได้ ตัวอย่างเช่น



นิพจน์นี้มีความหมายว่าอาจมีตัวอักษรใดก็ได้ ดังนั้นจะตรงกับข้อความที่มี หรือไม่มีตัวอักษรต่อท้ายก็ได้ ดังนั้นจึงสามารถตรงกับข้อความว่า สวัสดีจ้า สวัสดีค่ะ หรือสวัสดีก็ได้

**ส่วนหัว**

**ส่วนหัวมีไว้สำหรับการประกาศตัวแปรโดยส่วนหัวจะเริ่มต้นด้วยขีดกลาง - 3 ขีด และปิดด้วยขีดกลางอีก 3 ขีด โดยการกำหนดตัวแปรนั้นจะใช้เครื่องหมาย** <**-** **โดยการตั้งชื่อตัวแปรนั้น สามารถตั้งโดยใช้ตัวหรือตัวอักษรก็ได้แต่ห้ามขึ้นตั้นด้วยตัวเลข** **ตัวอย่างเช่น**

****

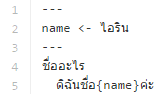
**โดยจากตัวอย่างโค้ดคือการประกาศตัวแปร 3 ตัว ตัวแปรแรกชื่อ** name **เก็บค่า ไอริน ตัวแปรที่ 2 ชื่อ** gender **เก็บค่าหญิง ตัวแปรที่ 3 เก็บค่า 18**

การเข้าถึงตัวแปร

**การเข้าถึงตัวแปรสามารถทำได้โดยการใช้เครื่องหมายปีกกา** { } **โดยแบ่งตัวแปรเป็น** 2 ประเภทได้แก่ตัวแปรทั่วไปและตัวแปรตามเหตุการณ์

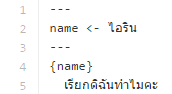
**ตัวแปรทั่วไป**

คือตัวแปรที่ประกาศไว้บนส่วนหัวของไฟล์ สามารถเรียกใช้งานได้โดยเขียนว่า {ชื่อตัวแปร} แล้วตัวแปลภาษาจะทำการแทนค่าของตัวแปรไปยังบริเวณที่ {ชื่อตัวแปร} ปรากฏอยู่ ตัวอย่างเช่น



เมื่อถามระบบว่า ชื่ออะไร ระบบจะตอบว่า ดิฉันชื่อไอรินค่ะ เนื่องจากตัวแปลภาษาได้แทนค่าของตัวแปร name ลงไปยังบริเวณที่เขียนว่า {name}

นอกจากนี้เรายังสามารถใช้ตัวแปรทั่วไปกับคำถามได้ด้วย ตัวอย่างเช่น



เมื่อเราถามระบบว่า ไอริน ระบบจะตอบกลับว่า เรียกดิฉันทำไมคะ เนื่องจากค่าของตัวแปร name ซึ่งมีค่าเป็นไอริน จะถูกแทนที่ลงในคำถาม ดังนั้นคำถามนั้นจึงเป็นคำว่า ไอริน ทำให้เมื่อถามว่าไอรินแล้วระบบจึงตอบว่า เรียกดิฉันทำไมคะ

**ตัวแปรตามเหตุการณ์**

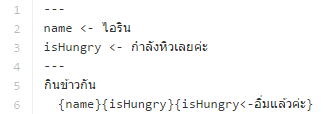
ตัวแปรตามเหตุการณ์จะเกิดขึ้นเมื่อมีการใช้นิพจน์ () และ \* เนื่องจากนิพจน์ 2 ตัวนี้จะต้องมีค่าใดค่าหนึ่งแน่นอน โดยการเข้าถึงจะใช้ {ตำแหน่งของนิพจน์} โดยนับเป็นเลขเริ่มต้นจาก 1 ตัวอย่างเช่น



เมื่อถามระบบว่า ผมชอบแมว ระบบจะตอบกลับมาว่า ทำไมคุณถึงชอบพวกแมวละเนื่องจากคำว่าชอบ ถูกนำไปแทนค่าในตัวแปร {1} และคำว่า แมว ถูกนำไปแทนค่าในตัวแปร {2}

**การเปลี่ยนค่าตัวแปร**

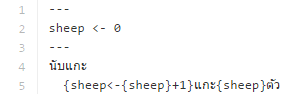
**เ**ราสามารถเปลี่ยนแปลงค่าของตัวแปรแบบทั่วไป ที่มีอยู่เดิมได้การกำหนดค่าตัวแปรเข้าไปใหม่แบบพลวัตโดยทำการใช้เครื่องหมาย <- เพื่อกำหนดค่าตัวแปรเข้ามาใหม่ภายใต้เครื่องหมายปีกกา ตัวอย่างเช่น



จากตัวอย่าง เมื่อถามระบบว่า กินข้าวกัน ระบบจะตอบกลับมาว่า ไอรินกำลังหิวเลยค่ะ เมื่อถามระบบอีกครั้งว่า กินข้าวกัน ระบบจะตอบกลับมาว่า ไอรินอิ่มแล้วค่ะ

โดยการกำหนดค่านั้นจะทำเมื่อเจอวงเล็บปีกกาที่มีเครื่องหมาย <- ดังนั้นหากเราสลับตำแหน่งของวงเล็บปีกกาในโค้ดตัวอย่างด้านเป็น {name}{isHungry<-อิ่มแล้วค่ะ}{isHungry} เมื่อถามว่ากินข้าวกัน ระบบจะตอบกลับมาว่า ไอรินอิ่มแล้วค่ะ เนื่องจากโค้ดดังกล่าวได้ทำการกำหนดค่าตัวแปรใหม่ก่อนนำไปแสดงผล

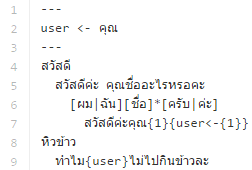
อีกทั้งภายใต้เครื่องหมายปีกกายังสามารถกำหนดค่าได้โดยการใช้เครื่องหมายปีกกา ยังสามารถใช้เครื่องหมาย บวก ลบ คูณ หาร ในการช่วยกำหนดค่าได้ ตัวอย่างเช่น



หากถามระบบว่านับแกะ ระบบจะตอบว่า แกะ1ตัว หากถามระบบอีกครั้งว่า นับแกะ ระบบจะตอบว่าแกะ 2 ตัวและเพิ่มขึ้นเรื่อยๆตามลำดับ

สังเกตว่าคำสั่ง {sheep<-{sheep}+1} จำเป็นต้องใส่วงเล็บปีกกาที่ชื่อตัวแปรในฝั่งขวาของเครื่องหมาย <- เสมอ หากไม่ใส่เครื่องหมายปีกการะบบจะถือว่าคำว่า sheep เป็นอักขระ ไม่ใช่ตัวแปร

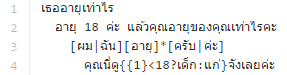
และเรายังสามารถนำค่าที่ได้มาจากตัวแปรตามเหตุการณ์กำหนดค่าใส่ในตัวแปรทั่วไปเพื่อใช้งานค่าในภายหลังได้ ดังตัวอย่าง



เมื่อถามระบบว่า หิวข้าว ระบบจะตอบว่า ทำไมคุณไม่ไปกินข้าวละ เมื่อถามระบบต่อว่า สวัสดี ระบบจะตอว่า สวัสดีค่ะ คุณชื่ออะไรหรอคะ จึงถามระบบต่อว่า ผมชื่อเพียวครับ ระบบตอบกลับมาว่า สวัสดีค่ะคุณเพียว และเมื่อถามระบบต่อว่า หิวข้าว ระบบจะตอบว่า ทำไมเพียวไม่ไปกินข้าวละ จะสังเกตว่าคำถามว่าหิวข้าวทั้ง 2 ครั้งชื่อผู้ใช้จะแตกต่างกัน

**เงื่อนไขในบรรทัด**

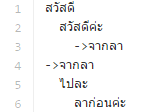
เราสามารถทำให้คำตอบการสนทนาเปลี่ยนไปได้โดยการใช้เงื่อนไขในบรรทัดได้ โดยใช้เครื่องหมาย ? ตามด้วยคำตอบเมื่อเป็นจริง แบ่งด้วย : แล้วตามด้วยเงื่อนไขเมื่อเป็นเท็จ โดยเครื่องหมายในการเปรียบเทียบที่สามารถใช้ได้ ได้แก่ และ (&&), หรือ (||), นิเศษ (!), เท่ากับ (==), มากกว่าเท่ากับ (>=), น้อยกว่าเท่ากับ (<=), มากกว่า (>), น้อยกว่า (<), บวก(+), ลบ (-), คูณ (\*) และหาร (/) ตัวอย่างเช่น



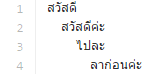
เมื่อถามระบบว่า เธออายุเท่าไร ระบบจะตอบว่า อายุ 18 ค่ะ แล้วคุณอายุของคุณเท่าไรคะ เมื่อถามกลับไปว่า อายุ 20 ครับ ระบบจะตอบกลับมาว่า คุณนี่ดูแก่จังเลยค่ะ แต่หากถามกลับไปว่า อายุ 12 ครับ ระบบจะตอบว่า คุณนี่ดูเด็กจังเลยค่ะ

**หัวข้อเรื่อง**

ในภาษาไอริน ฟังก์ชันนั้นจะใช้เครื่องหมาย -> แทนการกำหนดหัวข้อเรื่อง ทั้งการประกาศหัวข้อเรื่องและการนำเข้าสู่หัวข้อเรื่อง โดยการประกาศและใช้หัวข้อเรื่องจะ ->ชื่อหัวข้อ ต่างกันที่การประกาศหัวข้อเรื่องนั้นนั้นก่อนหน้าเครื่องหมาย -> จะไม่มีการเยื้องเลย หากมีการเยื้องจะเป็นการนำเข้าสู่หัวข้อเรื่องนั้น ตัวอย่างเช่น



เมื่อถามระบบว่า สวัสดี ระบบจะตอบว่า สวัสดีค่ะ หลังจากนั้นถามว่า ไปละ ต่อระบบจะตอบว่า ลาก่อนค่ะ เนื่องจากเยื้องของคำว่าสวัสดีค่ะ มีการเรียกใช้หัวข้อจากลา ซึ่งจะเป็นการดึงเอาคำถามและคำตอบที่อยู่ในหัวข้อจากลาไปเชื่อมอยู่กับคำถามสวัสดี ดังนั้นคำถามที่อยู่ต่อจากคำว่าสวัสดีค่ะก็คือคำว่าไปละ หรือก็คือผลลัพธ์ที่ได้จะเหมือนกับการเขียนโค้ดดังนี้



และยังสามารถเรียกหัวข้อจากไฟล์อื่นได้โดยการนำหัวข้อเรื่องใส่ไว้ในไฟล์อื่นแล้วตั้งชื่อไฟล์ด้วยนามสกุล .irin แล้วทำการเรียกใช้ด้วยคำสั่งหัวข้อเรื่อง

ตัวอย่างเช่น ไฟล์แรกเขียนว่า



ไฟล์ goodbye.irin เขียนว่าให้ทำการเขียนประโยคการโต้ตอบได้เลยโดยไม่ต้องใส่ชื่อหัวข้อเรื่อง

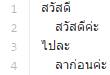


จะเห็นได้ว่าเมื่อรันแล้วผลลัพธ์จะสามารถใช้งานได้เหมือนกับการเขียนหัวข้อเรื่องไว้ในไฟล์เดียวกัน

แต่หากการเขียนเครื่องหมาย -> โดยไม่มีการเยื้องและตามด้วย .irin นั้นถือเป็นการนำเข้าไฟล์มาใช้งานไม่ใช่การประกาศหัวข้อเรื่อง ตัวอย่างเช่น

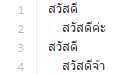


ผลลัพธ์ที่ได้จะทำงานเหมือนกับโค้ดดังต่อไปนี้



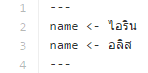
**บนลงล่าง**

สำหรับภาษาไอรินจะมีการเรียงลำดับความสัมพันธ์โดยการทำงานโค้ดจากบนลงล่าง หากโค้ดที่มีลำดับการเยื้องเดียวกันจะทำการเลือกโค้ดที่อยู่ด้านบนเสมอ ตัวอย่างเช่น



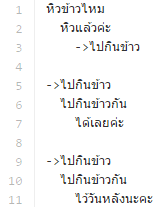
เมื่อถามระบบว่าสวัสดี ระบบจะตอบว่าสวัสดีค่ะ เสมอเนื่องจากสวัสดีในบรรทัดที่หนึ่งเป็นบรรทัดแรกที่ตรงกับเงื่อนไข ดังนั้นจึงตอบว่าสวัสดีค่ะเสมอ

นอกจากนี้การทำงานจากบนลงล่างยังมีผลต่อการกำหนดค่าตัวแปรอีกด้วยตัวอย่างเช่น



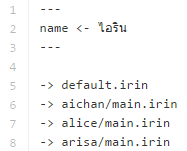
จากโค้ดตัวอย่างมีการประกาศตัวแปรชื่อ name ซ้ำกัน 2 ครั้งดังนั้นแล้วค่าของตัวแปรจะเป็น ไอริน เนื่องจากโค้ดพบว่า

การกำหนดหัวข้อชื่อซ้ำกันจตัวแปลภาษาจะเลือกใช้งานหัวข้อที่เจอเป็นหัวข้อแรก ตัวอย่างเช่น



เมื่อถามระบบว่า หิวข้าวไหม ระบบตอบว่าหิวแล้วค่ะ เมื่อถามระบบต่อว่า ไปกินข้าวกัน ระบบจะตอบว่าได้เลยค่ะเสมอ เนื่องจากเจอหัวข้อเรื่องที่บรรทัดที่ 5 ก่อน

ประโยชน์ของการใช้งานโค้ดแบบบนลงล่างคือไม่สามารถเขียนทับตัวแปรที่ของไฟล์อื่นได้เหมาะกับกรณีที่ไฟล์ภาษาไอรินมาจากการเขียนโดยคนละคนกันตัวอย่างเช่น



ถ้าทั้ง 4 ไฟล์เขียนโดยคนละคนกันแล้วใช้ ในแต่ละไฟล์มีการประกาศ ตัวแปรชื่อ name เหมือนกัน ตัวแปรทั้งหมดนั้นจะถูกเปลี่ยนค่าเป็น ไอรินทันทีเพื่อการอำนวยความสะดวกในการเขียนโค้ดของท่าน

นอกจากนี้แล้วท่านยังสามารถอ่านเอกสารวิธีการใช้งานภาษาไอรินและตัวแปลภาษาไอริน เพิ่มเติมได้ที่ https://ภาษา.ไอริน.ไทย และสามารถลองใช้งานภาษาไอรินออนไลน์ได้ที่ https://ภาษา.ไอริน.ไทย/ลองใช้