



Hozo-Ontology Editor

ค่าย YG

สมาชิกกลุ่ม11

623020368-0	นางสาวณัฐริกา แสนแก้ว	sec1
623020379-5	นางสาวบุญยวีร์ เลื่อนศรีอินญากุล	sec1
623020381-8	นางสาวปิยะชล ชื่นนิรันดร์	sec1
623020385-0	นางสาวพรทิพา ชนะน้อย	sec1
623020696-3	นางสาวสียาภรณ์ ศรีบุรินทร์	sec1

เสนอ

รศ.ดร.งามนิจ อัจฉินทร์

รายงานเล่มนี้เป็นส่วนหนึ่งของรายวิชา 342436 เทคโนโลยีเว็บแบบสื่อความหมาย

ภาคการศึกษาที่ 2 ปีการศึกษา 2564

สำนักวิชาศึกษาทั่วไป มหาวิทยาลัยขอนแก่น

คำนำ

รายงานเล่มนี้เป็นส่วนหนึ่ง342436 เทคโนโลยีเว็บแบบสื่อความหมาย โดยมีจุดประสงค์เพื่อพัฒนาออนไลน์เกี่ยวกับ
ค่าย YG สามารถใช้เครื่องมือ HOZO เพื่อถ่ายทอดและจัดเก็บองค์ความรู้ในรูปแบบของออนไลน์ได้ รวมถึงสามารถ
ประยุกต์ใช้คุณสมบัติต่างๆที่เป็นคุณสมบัติของ OWL ได้

ทั้งนี้เนื้อหาและข้อมูลต่างๆได้มีการศึกษาและรวบรวมข้อมูลจากเว็บไซต์ต่างๆ ผู้จัดทำต้องขอขอบคุณ รศ.ดร.งามนิจ
อาจอินทร์ ผู้ให้ความรู้และแนวทางการศึกษา ผู้จัดทำหวังว่ารายงานฉบับนี้จะให้ความรู้ และเป็นประโยชน์ต่อผู้อ่านทุกท่าน
หากมีข้อผิดพลาดประการใด คณะผู้จัดทำขอน้อมรับไว้ และขออภัยมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

สารบัญ

เรื่อง	หน้า
1. รายละเอียดเครื่องมือที่ศึกษา	1
1.1 เครื่องมือที่ใช้ออกแบบออนโทโลยี : Hozo-Ontology Editor	1
1.2 ความสำคัญของเครื่องมือที่ศึกษา	1
1.3 ขอบเขตและฟังก์ชันการทำงานของเครื่องมือ	1
1.4 ข้อดีและข้อเสีย(ข้อจำกัด)ของเครื่องมือ	1
1.5 โครงสร้างออนโทโลยี	2
1.6 คุณสมบัติของ Web Ontology Language (OWL)	3
2. รายละเอียดเครื่องมือที่ศึกษา	4
2.1 Data Dictionary (Class)	4
2.2 Data Dictionary (DataProperty)	4
2.3 Data Dictionary (ObjectProperty)	5
3. คุณสมบัติOWL ที่นักศึกษาเลือกใช้และต้องอธิบายอย่างละเอียด	6
3.1 คุณสมบัติ Inverse Properties	6
3.2 คุณสมบัติ Cardinality Constraint	6
3.3 คุณสมบัติ disjointWith	7
4. ฟังก์ชันการทำงานของเครื่องมือ	7
4.1 วิธีการสร้างโปรเจกต์	7
4.2 วิธีการสร้างออนโทโลยีใหม่ (New Ontology)	8
4.3 วิธีการสร้างโหนดและซับโหนด (Add Node)	9
4.4 วิธีการเพิ่ม Slot (Add Slot)	11
4.5 วิธีการสร้าง Instance	13
4.6 วิธีการสร้าง Instance	14
4.7 วิธีการบันทึกไฟล์ (Save File)	15
4.8 วิธีการ Export ไฟล์	16

เรื่อง	หน้า
5.คู่มือการติดตั้งเครื่องมือและวิธีการใช้งานอย่างละเอียด	18
5.1. วิธีการบันทึกไฟล์ (Save File)	18
5.2.วิธีการสร้างโปรเจกต์	22
5.3.วิธีการสร้างออนโทโลยีใหม่ (New Ontology)	23
5.4.วิธีการสร้างโหนดและซับโหนด (Add Node)	24
5.5.วิธีการเพิ่ม Slot (Add Slot)	26
5.6.วิธีการลบวัตถุ (Deleting Object)	31
5.7.วิธีการสร้าง Instance	32
5.8.วิธีการบันทึกไฟล์ (Save File)	33
5.9.วิธีการ Export ไฟล์	33
6.ความต้องการทางด้าน Hardware และ Software ของเครื่องมือ	35
อ้างอิง	36

1. รายละเอียดของเครื่องมือที่ศึกษา

1.1. เครื่องมือที่ใช้ออกแบบออนโทโลยี : Hozo-Ontology Editor



รูปที่1 สัญลักษณ์Hozo-Ontology Editor

1.2. ความสำคัญของเครื่องมือที่ศึกษา

โปรแกรม Hozo-Ontology Editor ที่พัฒนาขึ้นโดยมหาวิทยาลัยโอซากา (Osaka University) ประเทศญี่ปุ่น ในปัจจุบันได้รับความนิยม ซึ่งสามารถทำการจัดเก็บองค์ความรู้ในรูปแบบของออนโทโลยีได้อย่างสะดวกและไม่เสียค่าใช้จ่ายในการนำมาประยุกต์ใช้

ออนโทโลยี คือ การรวมแนวคิดสำคัญที่จำเป็นในการใช้อธิบายเป้าหมายของสิ่งหนึ่ง และแสดงความสัมพันธ์ระหว่างแนวคิดเหล่านี้ ความสัมพันธ์ขั้นพื้นฐานสูงสุด คือความสัมพันธ์แบบทั่วไป หรือความสัมพันธ์แบบจัดเป็น ซึ่งแสดงคำอธิบายแนวคิดโดยใช้ Label และชื่อลำดับชั้นด้วย is-a ซึ่งเป็นความสัมพันธ์พื้นฐานที่สุด

1.3. ขอบเขตและฟังก์ชันการทำงานของเครื่องมือ

- 1.3.1.1. กำหนดมโนทัศน์(concept)
- 1.3.1.2. กำหนดความสัมพันธ์ระหว่างมโนทัศน์แบบ"จัดเป็น" (IS-A)
- 1.3.1.3. กำหนดคุณสมบัติของมโนทัศน์(property)
- 1.3.1.4. ระบุข้อกำหนดของคุณสมบัติ (constraints)
- 1.3.1.5. Create Instances กำหนดค่าตัวแทนข้อมูลของ Conceptต่างๆ
- 1.3.1.6. สามารถ Export ไฟล์ได้หลายชนิดขึ้นอยู่กับเวอร์ชันของโปรแกรมที่ผู้ใช้ทำการดาวน์โหลด เช่น Text/HTML, CSV, RDF, OWL, DAML+OIL เป็นต้น

1.4. ข้อดีและข้อเสีย(ข้อจำกัด)ของเครื่องมือ

1.4.1. ข้อดี

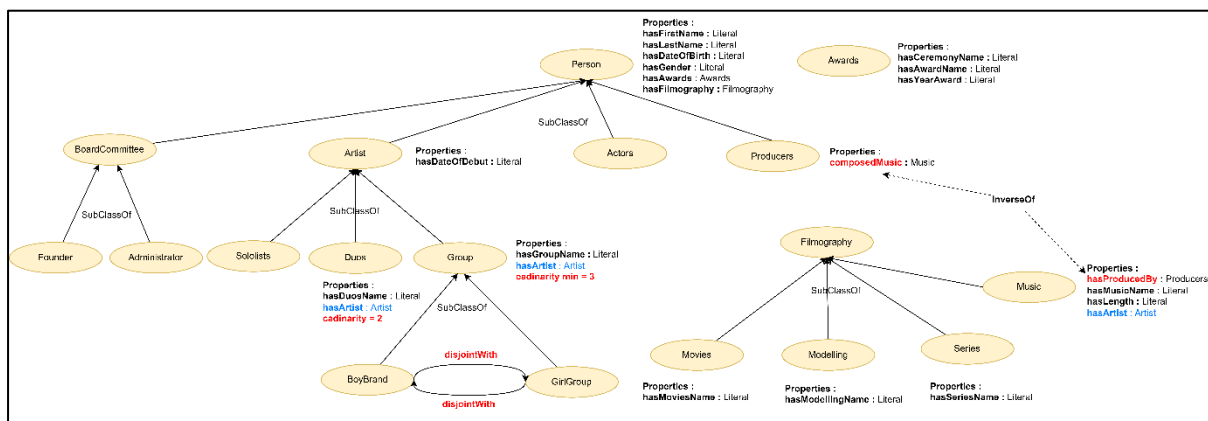
- 1.4.1.1. โปรแกรมสามารถใช้งานได้ฟรี ไม่มีค่าใช้จ่าย
- 1.4.1.2. GUI Tools ที่ช่วยให้การสร้าง Ontology ทำได้ง่าย ลากวางได้เลย เห็นภาพ และความสัมพันธ์ชัดเจน

1.4.2. ข้อเสีย

- 1.4.2.1. ยังไม่รองรับ Multi-Language
- 1.4.2.2. ถ้า Ontology ใหญ่มากเกินไป จะไม่สามารถแยกย่อยหลายๆ Ontology ได้

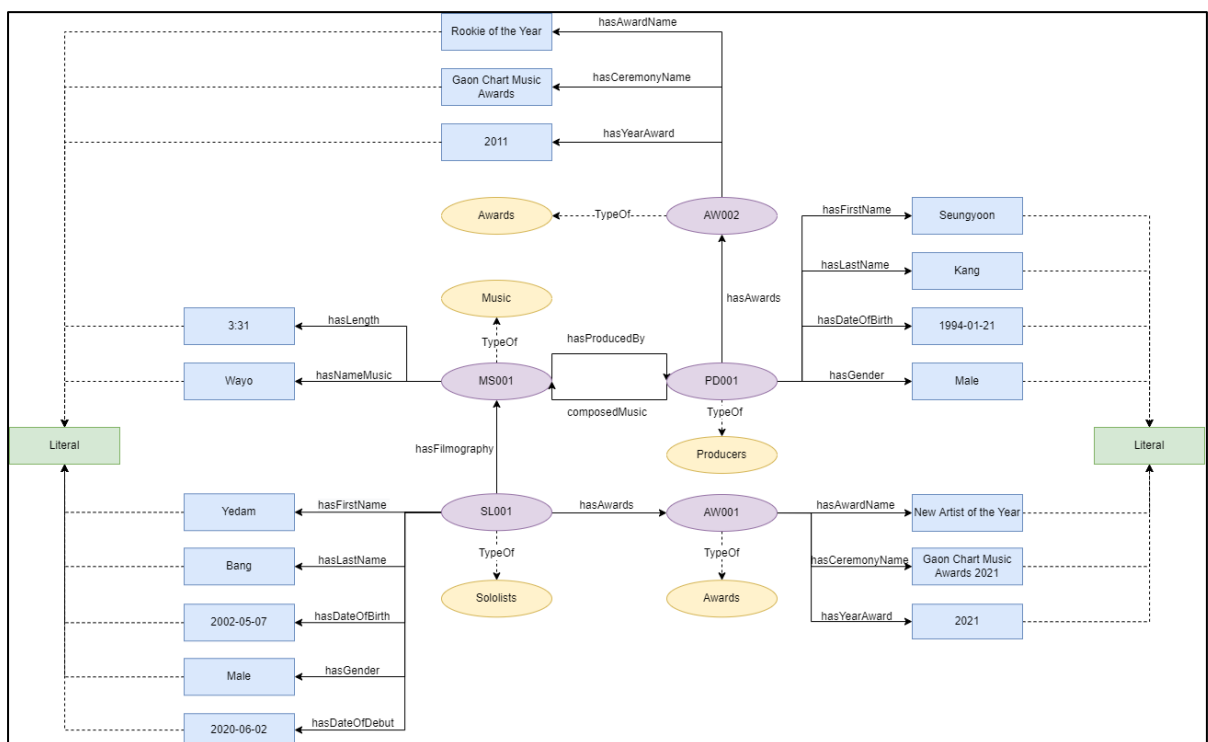
1.5. โครงสร้างออนไลน์

1.5.1. Schema Level



รูปที่ 2 Schema Level

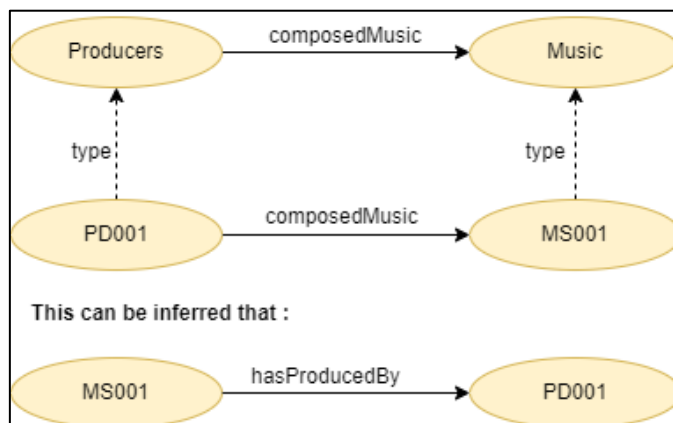
1.5.2. Instance Level



รูปที่ 3 Instance Level

1.6. คุณสมบัติของ Web Ontology Language (OWL)

1.6.1. คุณสมบัติ Inverse Properties



รูปที่4 คุณสมบัติ Inverse Properties

เนื่องจาก Property `composedMusic` ได้กำหนดให้เป็นคุณสมบัติแบบ Inverse Properties กับ Property `hasProducedBy` ทำให้คอมพิวเตอร์อนุมานได้ว่า Property `hasProducedBy` มี Domain เป็น Music และมี Range เป็น Producers

ตัวอย่างเช่น โปรดิวเซอร์ PD001 เป็นคนแต่งเพลงชื่อ MS001 ดังนั้น เราจะสามารถอนุมานได้ว่าเพลงที่ชื่อ MS001 ถูกแต่งโดยโปรดิวเซอร์ PD001

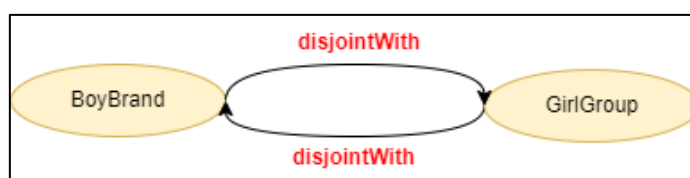
1.6.2. คุณสมบัติ Cardinality Constraints



รูปที่5 คุณสมบัติ Cardinality Constraints

เนื่องจาก Property `hasArtist` ได้กำหนดให้เป็นคุณสมบัติแบบ `minCardinality` จากตัวอย่างดังรูป ทำให้คอมพิวเตอร์อนุมานได้ว่าคลาส Group (กลุ่มศิลปิน) จะสามารถมีจำนวน Artist ได้ต่ำสุด 3 คนขึ้นไปไม่มีจำกัดจำนวนสูงสุด

1.6.3. คุณสมบัติ disjointWith



รูปที่6 คุณสมบัติ disjointWith

เนื่องจากคลาส BoyBrand และคลาส GirlGroup ได้กำหนดให้เป็นคุณสมบัติแบบ disjointWith ทำให้คอมพิวเตอร์อนุมานได้ว่าคลาส Instance ใดก็ตามที่เป็นสมาชิกของคลาส BoyBrand แล้วจะไม่สามารถเป็น Instance ของคลาส GirlGroup ได้

2. อธิบายในลักษณะ Data Dictionary

2.1. Data Dictionary (Class)

ตารางที่1 Data Dictionary (Class)

คลาสที่	ชื่อคลาส	ความหมาย	ซับคลาส
1	Person	คลาสแทนข้อมูลตำแหน่งหน้าที่ต่างๆในบริษัท	BoardCommittee, Artist, Actors, Producers
2	BoardCommittee	ซับคลาสของ Person แทนข้อมูลเกี่ยวกับบอร์ดผู้บริหาร	Founder, Administrator
3	Founder	ซับคลาสของ BoardCommittee แทนข้อมูลเกี่ยวกับผู้ก่อตั้ง	-
4	Administrator	ซับคลาสของ BoardCommittee แทนข้อมูลเกี่ยวกับผู้บริหาร	-
5	Artist	ซับคลาสของ Person แทนข้อมูลเกี่ยวกับศิลปิน	Sololists, Duos, Group
6	Sololists	ซับคลาสของ Artist แทนข้อมูลเกี่ยวกับศิลปินเดี่ยว	-
7	Duos	ซับคลาสของ Artist แทนข้อมูลเกี่ยวกับศิลปินคู่อี	-
8	Group	ซับคลาสของ Artist แทนข้อมูลเกี่ยวกับศิลปินกลุ่ม	BoyBrand, GirlGroup
9	BoyBrand	ซับคลาสของ Group แทนข้อมูลเกี่ยวกับศิลปินกลุ่มชาย	-
10	GirlGroup	ซับคลาสของ Group แทนข้อมูลเกี่ยวกับศิลปินกลุ่มหญิง	-

คลาสที่	ชื่อคลาส	ความหมาย	ชั้นคลาส
11	Actors	ชั้นคลาสของ Person แทนข้อมูลเกี่ยวกับนักแสดง	-
12	Producers	ชั้นคลาสของ Person แทนข้อมูลเกี่ยวกับโปรดิวเซอร์	-
13	Awards	คลาสข้อมูลของรางวัล	-
14	Filmography	คลาสแทนข้อมูลเกี่ยวกับผลงาน	Movies, Modelling, Series, Music
15	Movies	ชั้นคลาสของ Filmography แทนข้อมูลเกี่ยวกับผลงานภาพยนตร์	-
16	Modelling	ชั้นคลาสของ Filmography แทนข้อมูลเกี่ยวกับบอร์คผู้บริหาร	-
17	Series	ชั้นคลาสของ Filmography แทนข้อมูลเกี่ยวกับผลงานซีรีส์	-
18	Music	ชั้นคลาสของ Filmography แทนข้อมูลเกี่ยวกับผลงานเพลง	-

2.2. Data Dictionary (DataProperty)

ตารางที่2 Data Dictionary (DataProperty)

ชื่อคุณสมบัติ	ความหมายคุณสมบัติ	โดเมน	เรนจ์
hasFirstName	มีชื่อจริง	Person	Literal
hasLastName	มีนามสกุล	Person	Literal
hasDateOfBirth	มีวันเกิด	Person	Literal
hasGender	มีเพศ	Person	Literal
hasDateOfDebut	มีวันเดบิวต์	Artist	Literal
hasDuosName	มีชื่อศิลปินคู่โอ้	Duos	Literal

ชื่อคุณสมบัติ	ความหมายคุณสมบัติ	โดเมน	เรนจ์
hasGroupName	มีชื่อศิลปินกลุ่ม	Duos	Literal
hasMoviesName	มีชื่อภาพยนตร์	Movies	Literal
hasModellingName	มีชื่องานเดินแบบ	Modelling	Literal
hasSeriesName	มีชื่อซีรีส์	Series	Literal
hasMusicName	มีชื่อเพลง	Music	Literal
hasLength	มีความยาวเพลง	Music	Literal
hasCeremonyName	มีชื่องานประกาศรางวัล	Awards	Literal
hasAwardName	มีชื่อรางวัล	Awards	Literal
hasYearAward	มีปีที่ได้รับรางวัล	Awards	Literal

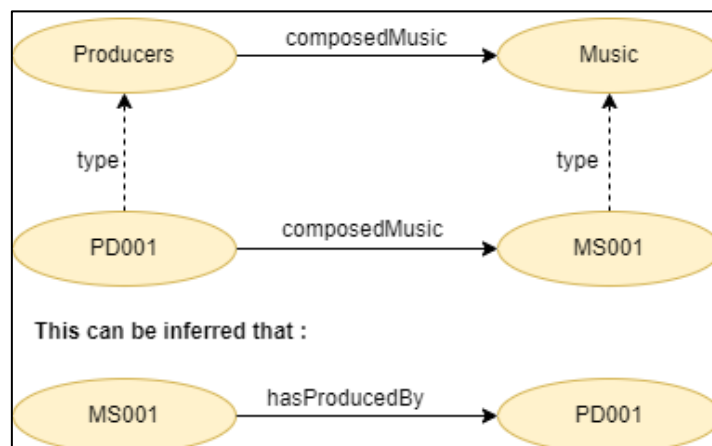
2.3. Data Dictionary (ObjectProperty)

ตารางที่2 Data Dictionary (ObjectProperty)

ชื่อคุณสมบัติ	ความหมายคุณสมบัติ	โดเมน	เรนจ์
hasAward	มีรางวัล	Person	Awards
hasFilmography	มีผลงาน	Person	Filmography
hasArtist	มีศิลปิน	Duos	Artist
hasArtist	มีศิลปิน	Group	Artist
hasProducedBy	มีการโปรดิวซ์โดย	Music	Producers
hasArtist	มีศิลปิน	Music	Artist
composedMusic	มีโปรดิวซ์เพลง	Producers	Music

3. คุณสมบัติOWL ที่นักศึกษาเลือกใช้และต้องอธิบายอย่างละเอียด

3.1. คุณสมบัติ Inverse Properties

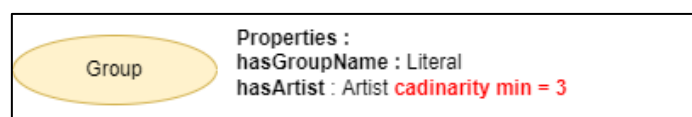


รูปที่7 คุณสมบัติ Inverse Properties

เนื่องจาก Property `composedMusic` ได้กำหนดให้เป็นคุณสมบัติแบบ Inverse Properties กับ Property `hasProducedBy` ทำให้คอมพิวเตอร์อนุมานได้ว่า Property `hasProducedBy` มี Domain เป็น Music และมี Range เป็น Producers

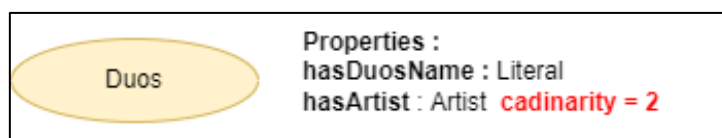
ตัวอย่างเช่น โปรดิวเซอร์ PD001 เป็นคนแต่งเพลงชื่อ MS001 ดังนั้น เราจะสามารถอนุมานได้ว่าเพลงที่ชื่อ MS001 ถูกแต่งโดยโปรดิวเซอร์ PD001

3.2. คุณสมบัติ Cardinality Constraints



รูปที่8 คุณสมบัติ Cardinality Constraints

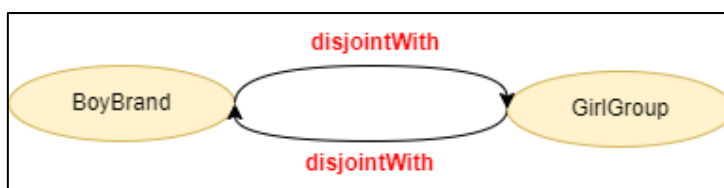
เนื่องจาก Property `hasArtist` ได้กำหนดให้เป็นคุณสมบัติแบบ `minCardinality` จากตัวอย่างดังรูป ทำให้คอมพิวเตอร์อนุมานได้ว่าคลาส Group (กลุ่มศิลปิน) จะสามารถมีจำนวน Artist ได้ต่ำสุด 3 คนขึ้นไปไม่มีจำกัดจำนวนสูงสุด



รูปที่9 คุณสมบัติ Cardinality Constraints

เนื่องจาก Property `hasArtist` ได้กำหนดให้เป็นคุณสมบัติแบบ Cardinality จากตัวอย่างดังรูป ทำให้คอมพิวเตอร์อนุมานได้ว่าคลาส Duos (ศิลปินคู่) จะสามารถมีจำนวน Artist ได้เพียง 2 คนเท่านั้น

3.3. คุณสมบัติ disjointWith




รูปที่ 10 คุณสมบัติ disjointWith

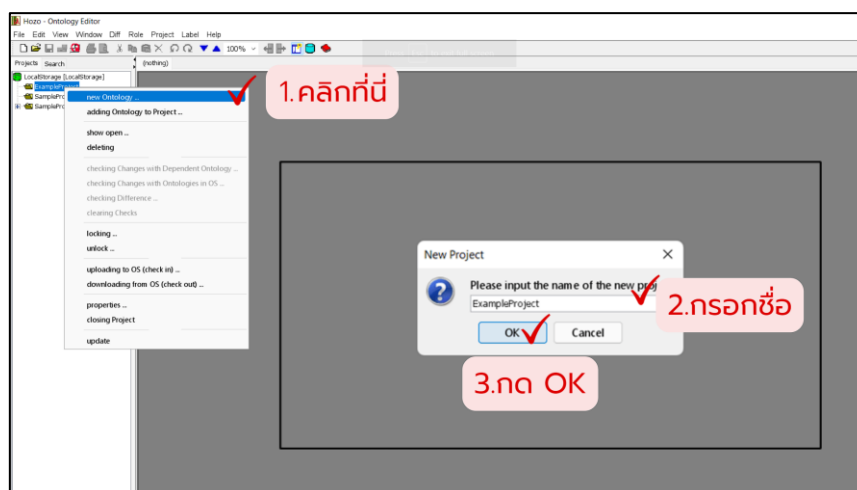
เนื่องจากคลาส BoyBrand และคลาส GirlGroup ได้กำหนดให้เป็นคุณสมบัติแบบ disjointWith ทำให้คอมพิวเตอร์อนุมานได้ว่าคลาส Instance ใดก็ตามที่เป็นสมาชิกของคลาส BoyBrand แล้วจะไม่สามารถเป็น Instance ของคลาส GirlGroup ได้

4. ฟังก์ชันการทำงานของเครื่องมือ

4.1. วิธีการสร้างโปรเจกต์

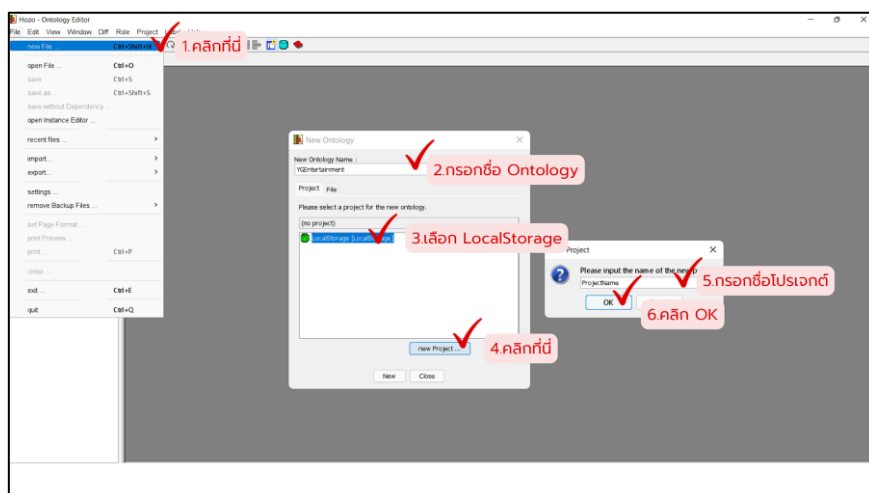
การสร้างโปรเจกต์ใหม่สามารถทำได้ 3 วิธีขึ้นอยู่กับความถนัดของผู้ใช้

- 4.1.1 คลิกขวาที่ LocalStorage [LocalStorage]  > new Project -> กรอกชื่อโปรเจกต์
-> กด OK เพื่อทำการสร้างโปรเจกต์



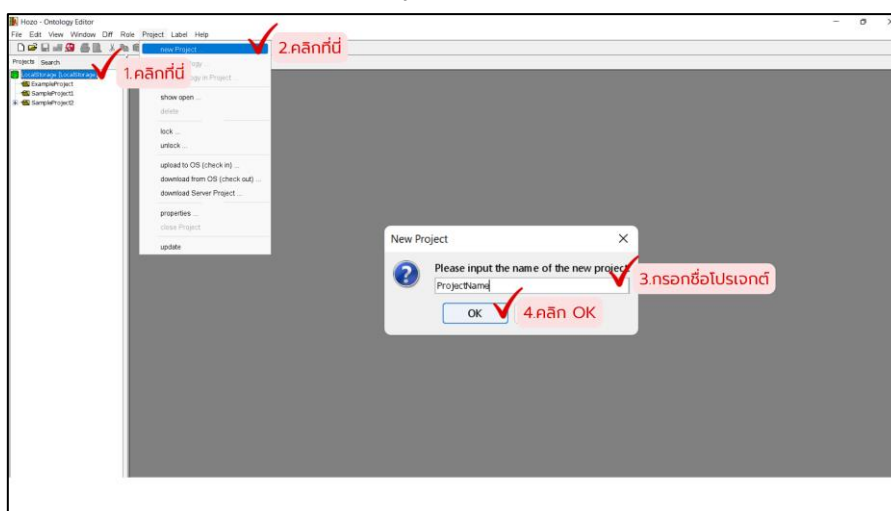
รูปที่ 11 วิธีการสร้างโปรเจกต์แบบที่ 1

- 4.1.2 คลิก File -> new File -> กรอกชื่อ Ontology -> เลือก LocalStorage [LocalStorage] -> New Project -> กรอกชื่อโปรเจกต์ วิธีนี้จะสามารถสร้าง Ontology ไปพร้อมกันได้เลย



รูปที่ 12 วิธีการสร้างโปรเจกต์แบบที่ 2

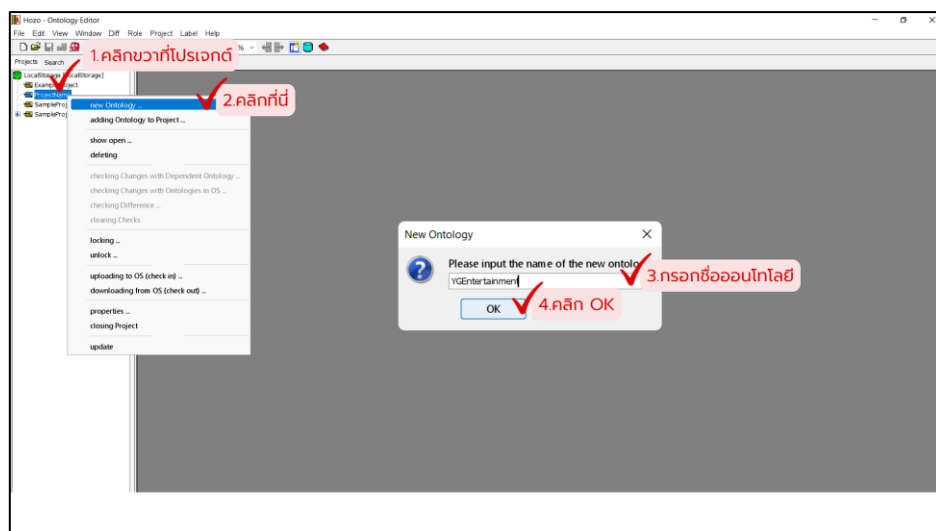
- 4.1.3 คลิก LocalStorage -> คลิกเมนู Project -> new Project -> กรอกชื่อโปรเจกต์ -> คลิก OK



รูปที่ 13 วิธีการสร้างโปรเจกต์แบบที่ 3

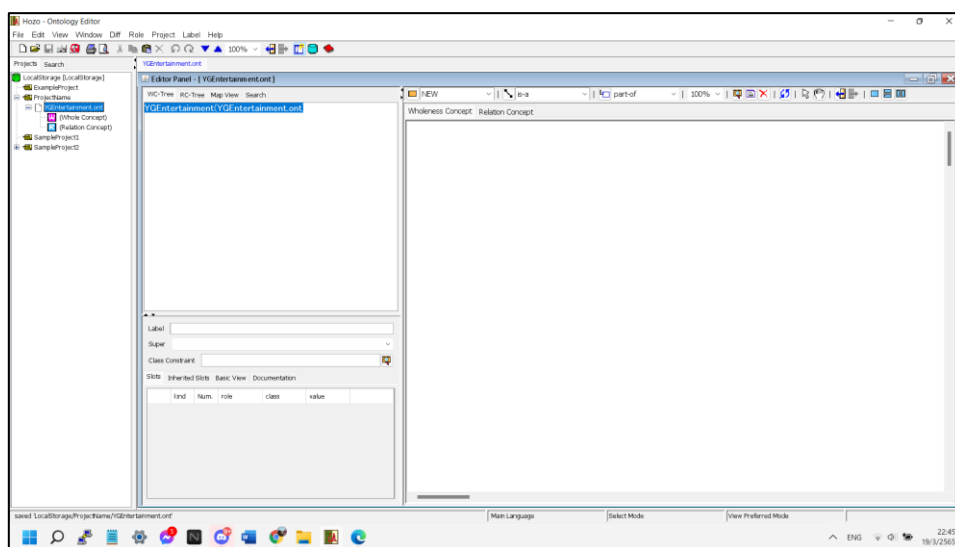
4.2. วิธีการสร้างออนโทโลยีใหม่ (New Ontology)

จากการสร้างโปรเจกต์ใหม่จะเห็นว่ามียังวิธีที่ 2 ที่มีการสร้างออนโทโลยีไปในขั้นตอนเดียวกันเลย ดังนั้นวิธีที่ 2 จึงไม่จำเป็นต้องสร้างออนโทโลยีใหม่อีกครั้ง มีเพียงวิธีที่ 1 และ 3 ที่ต้องดำเนินการต่อด้วยขั้นตอนนี้



รูปที่ 14 วิธีการสร้างออนโทโลยีใหม่

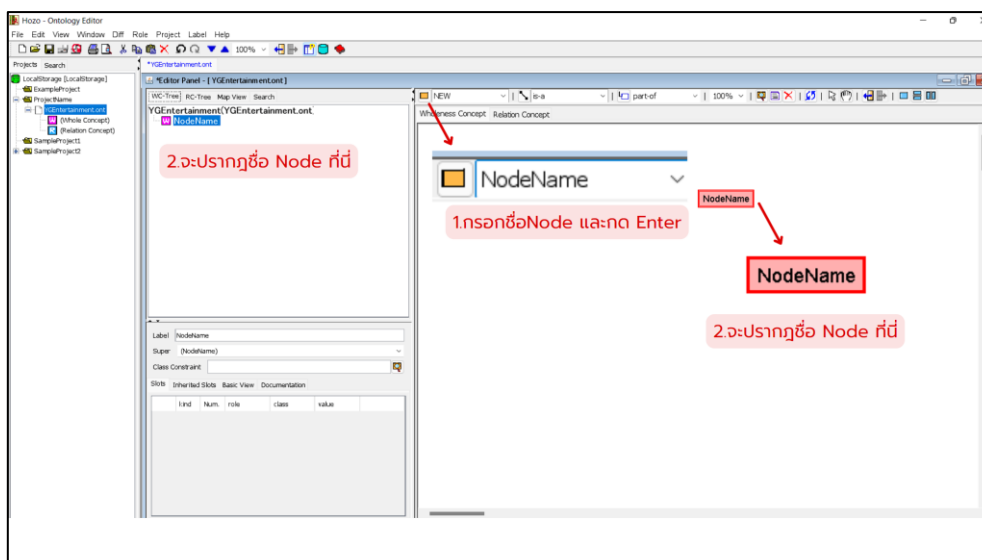
เมื่อสร้างเสร็จและดับเบิลคลิกไฟล์ออนโทโลยีที่เพิ่งสร้างจะด้ดังภาพ



รูปที่ 15 ภาพหน้าจอไฟล์ออนโทโลยี

4.3. วิธีการสร้างโหนดและซับโหนด (Add Node)

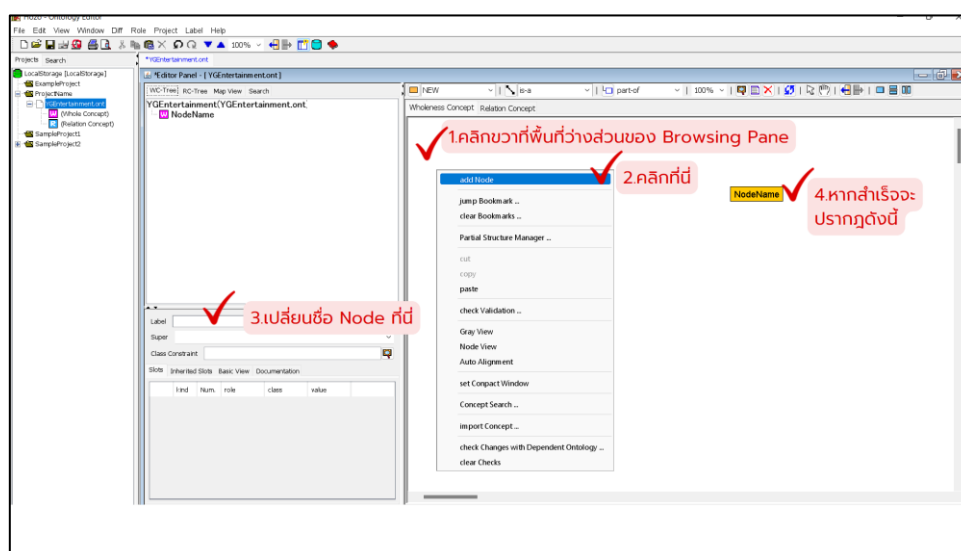
สามารถสร้างโหนด โดยการคลิกที่ไอคอนสร้างโหนด โดยจะสามารถสร้างได้ทีละ 1 โหนดโดยผู้ใช้งานสามารถเปลี่ยนชื่อโหนด ได้ตามต้องการ



รูปที่ 16 วิธีที่ 1 สำหรับการสร้างโหนด (Concept)

โดยอีกวิธีหนึ่งคือ คลิกขวาที่ส่วน Browsing Pane (ส่วนแสดงกราฟฟิก) จะปรากฏดังรูป จากนั้นคลิกตามขั้นตอน ดังนี้

add Node -> เปลี่ยนชื่อ โหนด จาก New เป็นชื่อตามที่ผู้ใช้งานต้องการ ผู้ใช้สามารถเปลี่ยนชื่อโหนด (Concept) ได้ที่ลาเบล (label) ในส่วนของ Definition Pane

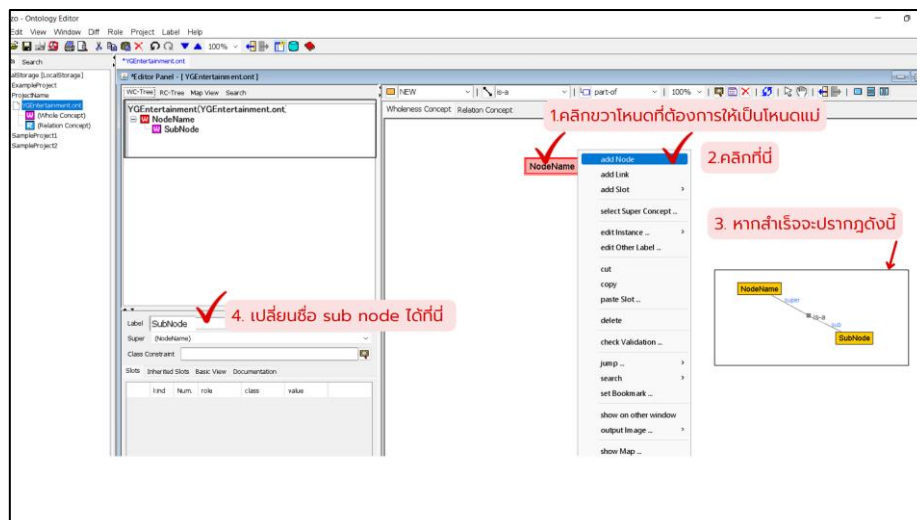


รูปที่ 17 วิธีที่ 2 สำหรับการสร้างโหนด (Concept)

หากผู้ใช้งานต้องการสร้างซับโนด (Sub Node) สามารถทำได้ 3 วิธีโดยมีวิธีดำเนินการขั้นตอนดังนี้

4.3.1. เลือก Node ที่ต้องการให้เป็นโนดแม่ -> คลิกขวา -> เลือก add Node

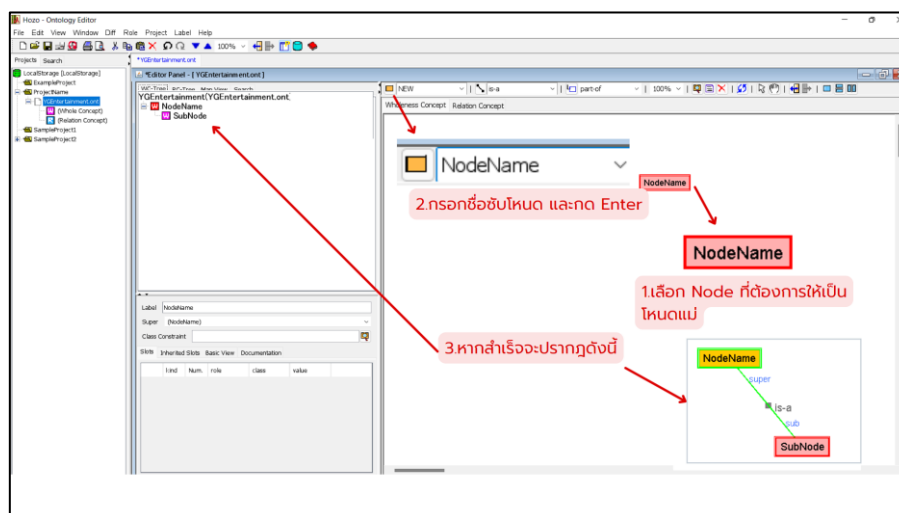
หากสำเร็จจะปรากฏดังรูป



รูปที่ 18 วิธีที่ 1 สำหรับการสร้างซับโนด

4.3.2. เลือก Node ที่ต้องการให้เป็นโนดแม่ -> คลิกที่ไอคอนสร้าง Node ในพื้นที่ Editing Tool Bar -&br> พิมพ์ชื่อ SubNode ที่ต้องการ

หากสำเร็จจะปรากฏดังรูป

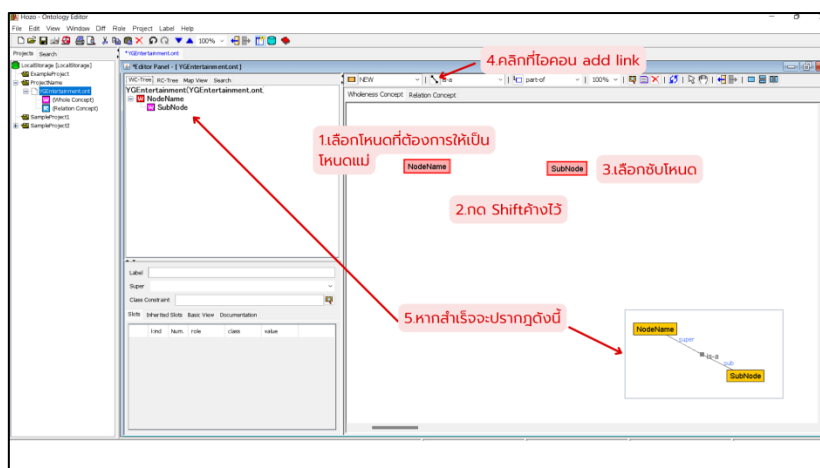


รูปที่ 19 วิธีที่ 2 สำหรับการสร้างซับโนด

4.3.3. วิธีนี้เป็นวิธีสร้างความสัมพันธ์ระหว่างสองโนดที่ไม่มีความสัมพันธ์กันมาก่อน

เลือก Node ที่ต้องการให้เป็นโนดแม่ -> กดปุ่ม Shift บนแป้นพิมพ์ค้างไว้ -> เลือก Node ที่ต้องการให้เป็น Sub Node -> คลิกที่ไอคอน add link เพื่อสร้างความสัมพันธ์ระหว่างโนด

หากสำเร็จจะปรากฏดังรูป



รูปที่ 20 วิธีที่ 3 สำหรับการสร้างซัพโหนด

4.4. วิธีการเพิ่ม Slot (Add Slot)

เป็นการกำหนดคุณสมบัติของโน้ตส์ (Property) หรือเป็นการกำหนดส่วนประกอบหรือคุณสมบัติของ Node นั้นๆ โดยประเภทความสัมพันธ์ มี 2 ประเภทดังนี้

1. part-of เป็นการกำหนดส่วนประกอบ
2. attribute-of เป็นการกำหนดคุณลักษณะ

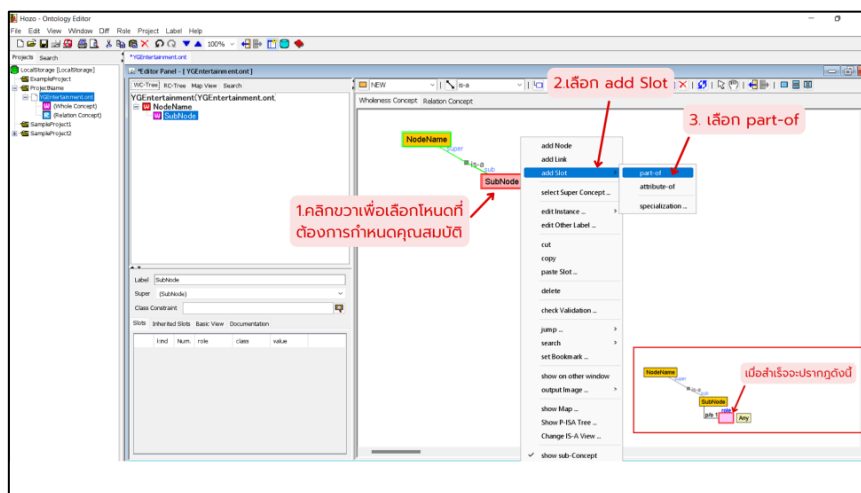
โดยจะยกตัวอย่างออนโทโลยีดอกไม้ เช่น

1. part-of ได้แก่ ดอก, ใบ, กิ่ง, ก้าน, ลำต้น, ราก เป็นต้น
2. attribute-of ได้แก่ สายพันธุ์, สีของดอก เป็นต้น

ในการเพิ่มคุณสมบัติภายในโปรแกรมมีขั้นตอนดังนี้

4.4.1. การเพิ่มคุณสมบัติแบบ part-of

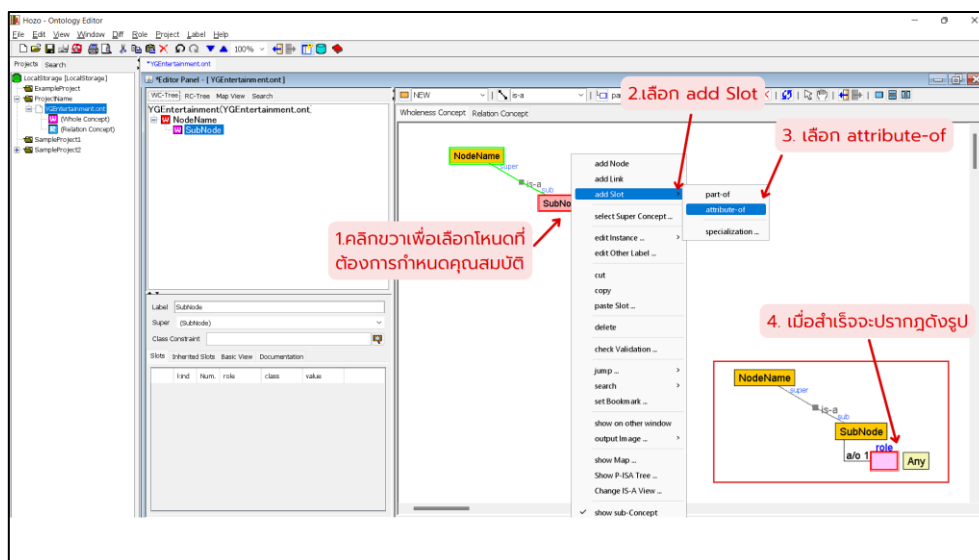
คลิกขวาเพื่อเลือกโหนดที่ต้องการกำหนดคุณสมบัติ -> เลือก add Slot -> เลือก part-of



รูปที่ 21 วิธีเพิ่มคุณสมบัติแบบ part-of

4.4.2. การเพิ่มคุณสมบัติแบบ attribute-of

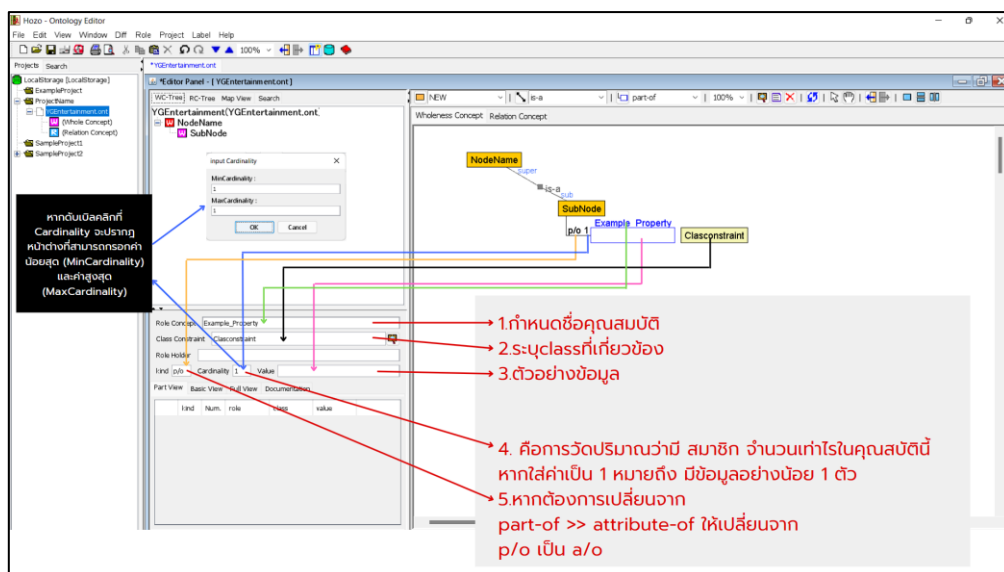
คลิกขวาเพื่อเลือกโหนดที่ต้องการกำหนดคุณสมบัติ -> เลือก add Slot -> เลือก attribute -of



รูปที่ 22 วิธีเพิ่มคุณสมบัติแบบ attribute -of

จากนั้นผู้ใช้งานสามารถกำหนดรายละเอียดขององค์ประกอบเบื้องต้นได้ดังนี้

1. Role Concept - กำหนดชื่อคุณสมบัติ
2. Class Constraint - ระบุคลาสที่เกี่ยวข้อง
3. Value - ตัวอย่างข้อมูล
4. Cardinality - คือการวัดปริมาณว่ามี สมาชิก จำนวนเท่าไรในคุณสมบัตินี้ หากใส่ค่าเป็น 1 หมายถึง มีข้อมูลอย่างน้อย 1 ตัว
5. Kind - หากต้องการเปลี่ยนจาก part-of -> attribute-of ให้เปลี่ยนจาก p/o เป็น a/o

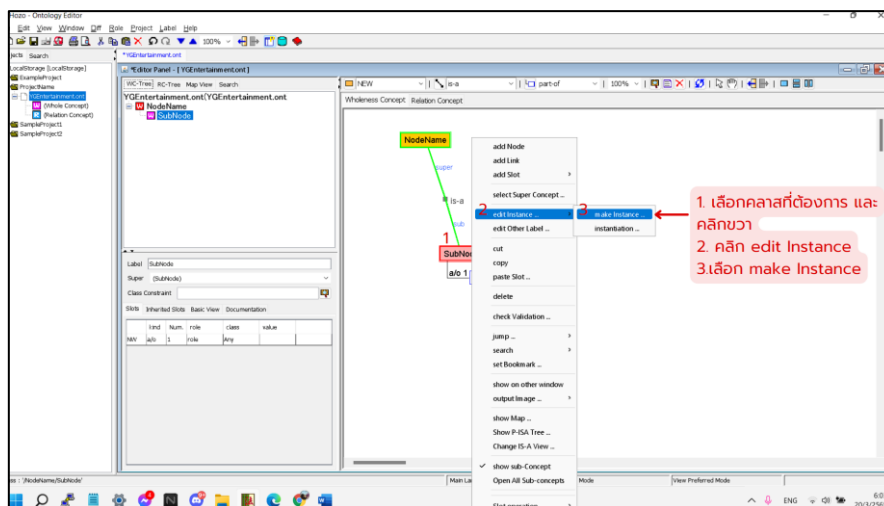


รูปที่ 23 วิธีการกำหนดคุณสมบัติแบบ part-of/attribute-of

4.5. วิธีการสร้าง Instance

เป็นการสร้างข้อมูลตัวอย่าง หลังจากที่มีการสร้าง Schema Level แล้ว เราก็ควรจะมีการแสดงข้อมูลตัวอย่างหรือ Instance Level เพื่อให้เข้าใจใน Ontology ที่สร้างมากยิ่งขึ้น วิธีการสร้าง Instance มีดังนี้

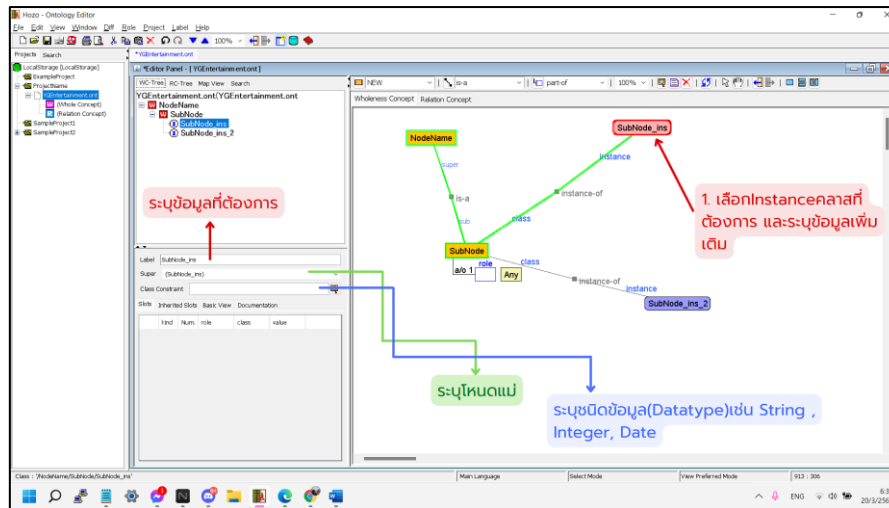
คลิกขวาคลาสที่ต้องการ -> เลือก edit Instance -> เลือก Make Instance



รูปที่ 24 วิธีสร้าง Instance

จากนั้นจะแสดงดังรูป และทำการระบุข้อมูลของ Instance class ที่เพิ่มเข้ามาโดย

เลือกInstanceคลาสที่ต้องการ และระบุข้อมูลเพิ่มเติม -> ระบุ label -> ระบุ -> Super Class -> ระบุ Class Constraint

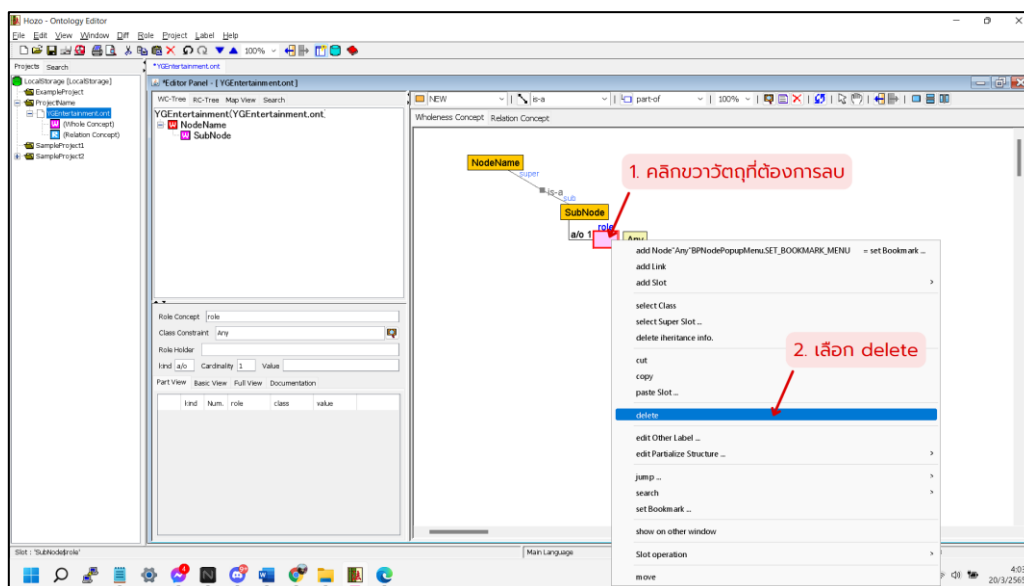


รูปที่ 25 วิธีสร้าง Instance

4.6. วิธีการสร้าง Instance

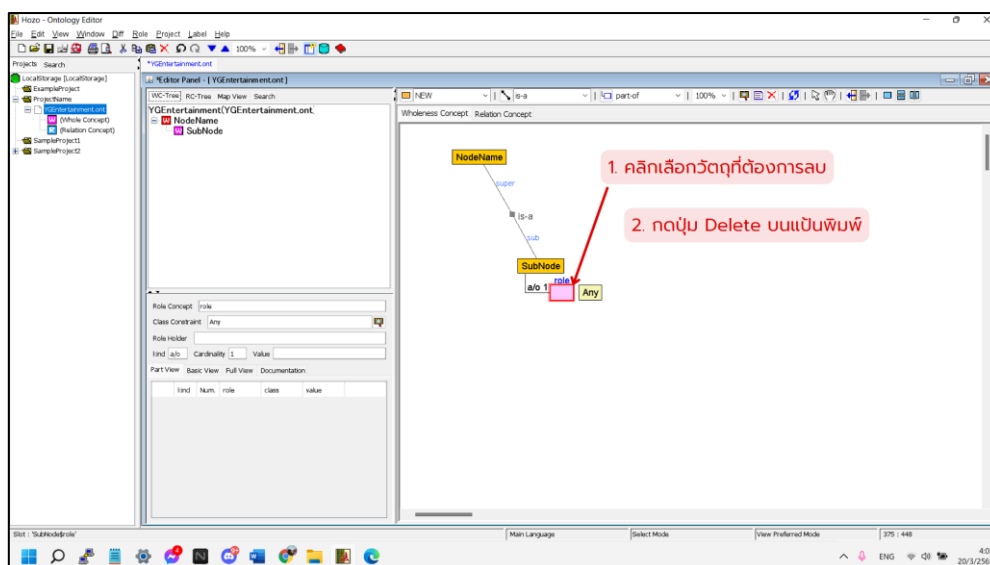
ผู้ใช้งานสามารถลบวัตถุได้ 2 วิธี คือ

4.6.1. คลิกขวาที่วัตถุ ที่ต้องการลบ -> เลือก Delete



รูปที่ วิธีที่ 26 สำหรับการลบวัตถุ

4.6.2. คลิกเลือกวัตถุ ที่ต้องการลบ -> กดปุ่ม Delete บนแป้นพิมพ์



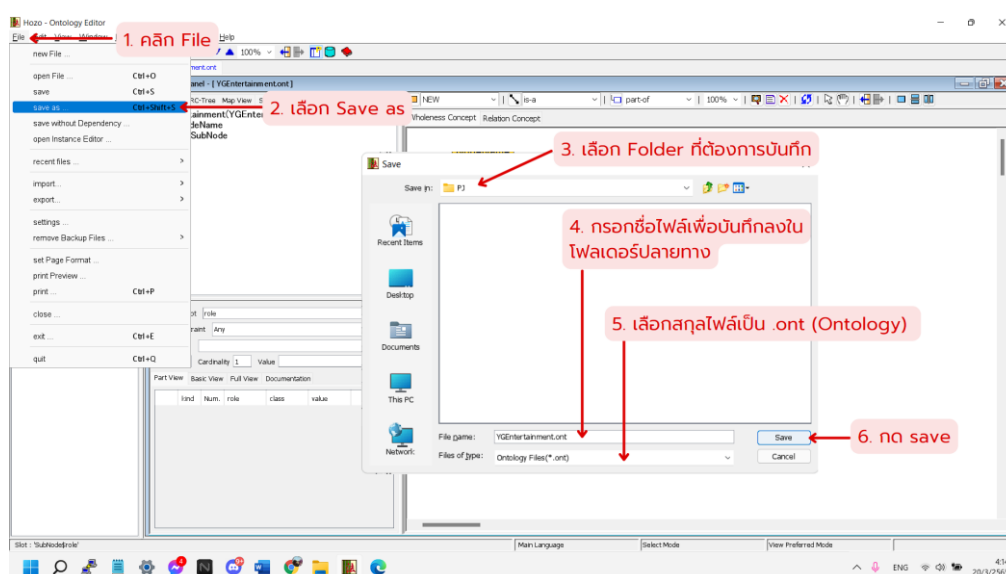
รูปที่ 27 วิธีที่ 2 สำหรับการลบวัตถุ

4.7. วิธีการบันทึกไฟล์ (Save File)

หากเป็นการบันทึกไฟล์ครั้งแรก ผู้ใช้งานสามารถบันทึกไฟล์ตามขั้นตอนดังนี้

คลิก File -> เลือก Save as -> เลือก Folder ที่ต้องการบันทึก -> กรอกชื่อไฟล์เพื่อบันทึกลงในโฟลเดอร์ปลายทาง

-> เลือกสกุลไฟล์เป็น .ont (Ontology) -> กด save

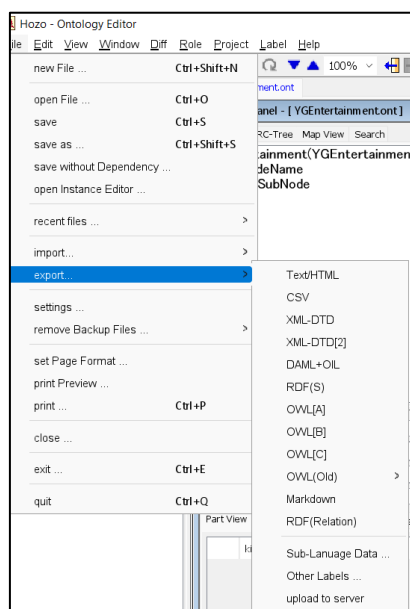


รูปที่ 28 วิธีการบันทึกไฟล์

4.8. วิธีการ Export ไฟล์

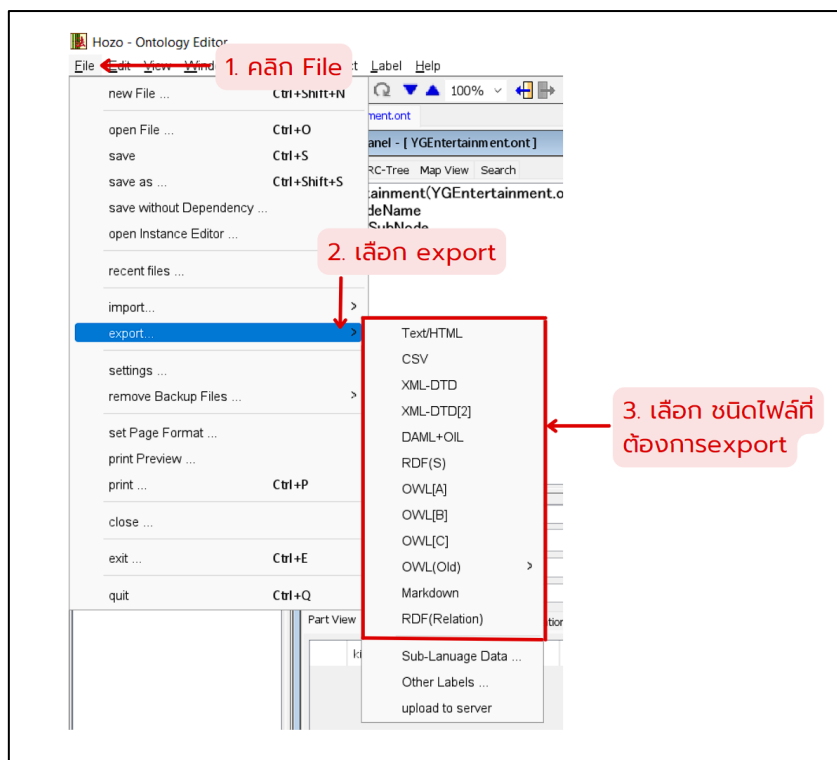
ผู้ใช้งานสามารถ export ไฟล์ออนโทโลยีเพื่อสามารถนำไปใช้งานได้สะดวกขึ้น ซึ่งโปรแกรมสามารถให้ผู้ใช้งาน export ไฟล์ได้หลายชนิด ดังนี้

1. Text/HTML
2. CSV
3. XML-DTD
4. XML-DTD[2]
5. DAML+OIL
6. RDF(S)
7. OWL[A]
8. OWL[B]
9. OWL[C]
10. OWL(Old)
 - 10.1 OWL
 - 10.2 OWL(2)
 - 10.3 OWL(3)
11. Markdown
12. RDF(Relation)



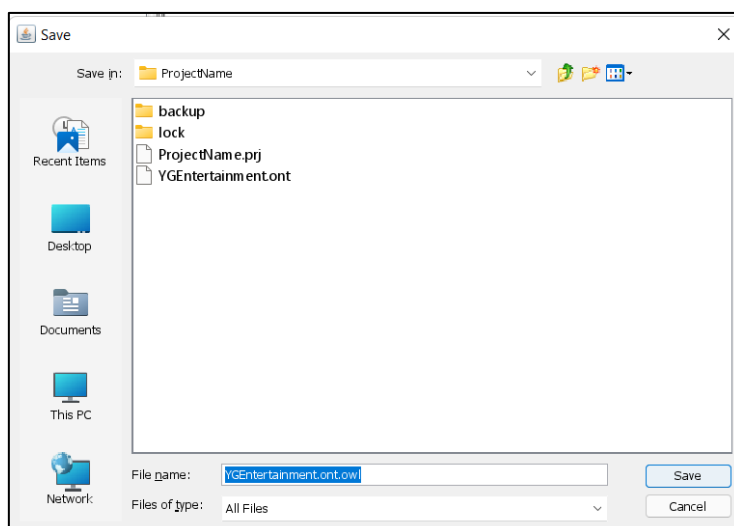
รูปที่ 29 ตัวอย่างชนิดไฟล์ของการ export

ผู้ใช้งานสามารถ export ไฟล์ออนโทโลยีตามขั้นตอน ดังนี้ คลิก File -> Export -> ชนิดไฟล์ที่ต้องการ Export



รูปที่ 30 วิธีการ export ไฟล์

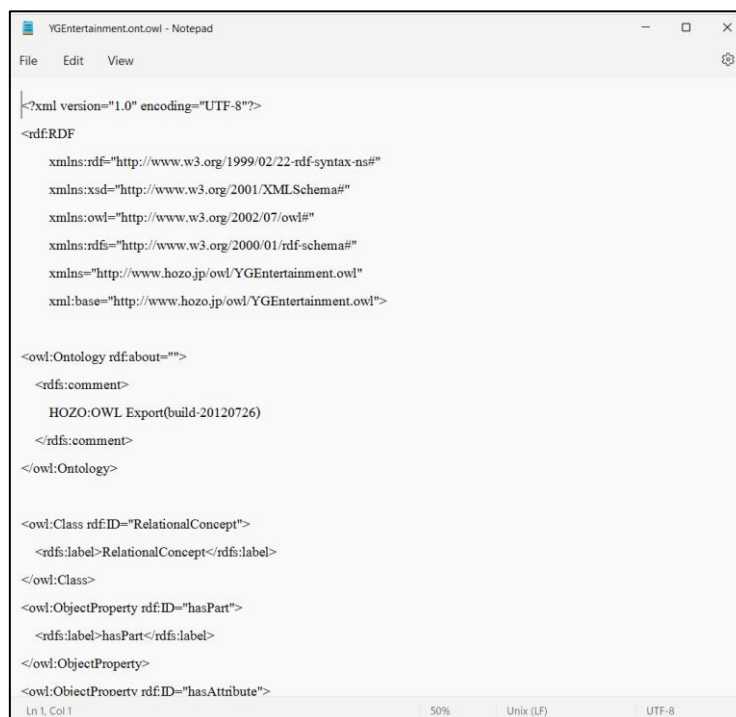
จากนั้นโปรแกรมจะแสดงหน้าต่างหนึ่ง หลักจากเลือกรูปแบบชนิดไฟล์ที่ต้องการ export ตัวอย่างเช่น ทำการ export file ในรูปแบบไฟล์ OWL[A] จะปรากฏดังรูป จากนั้นเลือกที่อยู่ของไฟล์เพื่อทำการบันทึกไว้ในโฟลเดอร์ที่ต้องการ



รูปที่ 31 เลือกที่อยู่ไฟล์เพื่อบันทึก

จากนั้นทำการทดลองเปิดไฟล์ OWL ในโปรแกรม NotePad ก็พบว่า จากที่ทำในโปรแกรมจะเป็นเพียงการสร้าง class ไม่จำเป็นต้องมีการเขียนโค้ด ทำให้เกิดความผิดพลาดน้อยและง่ายต่อการเข้าใจ เนื่องจากรูปแบบในการมองเห็นอยู่ใน

รูปแบบของ Class และ Subclass เมื่อผู้ใช้งานทำการ Export file ในรูปแบบของไฟล์อื่นๆ ข้อมูลภายในโปรแกรม HOZO ก็จะถูกแปลงเป็นวิธีเขียนของภาษานั้นๆ



```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.hozo.jp/owl/YGEntertainment.owl"
  xml:base="http://www.hozo.jp/owl/YGEntertainment.owl">

  <owl:Ontology rdf:about="">
    <rdfs:comment>
      HOZO:OWL Export(build-20120726)
    </rdfs:comment>
  </owl:Ontology>

  <owl:Class rdf:ID="RelationalConcept">
    <rdfs:label>RelationalConcept</rdfs:label>
  </owl:Class>

  <owl:ObjectProperty rdf:ID="hasPart">
    <rdfs:label>hasPart</rdfs:label>
  </owl:ObjectProperty>

  <owl:ObjectProperty rdf:ID="hasAttribute">

```

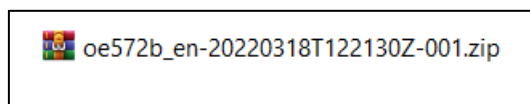
รูปที่ 32 ตัวอย่างข้อมูลไฟล์ OWL[A]

5. คู่มือการติดตั้งเครื่องมือและวิธีการใช้งานอย่างละเอียด

5.1. การติดตั้งและวิธีใช้งาน (Installation)

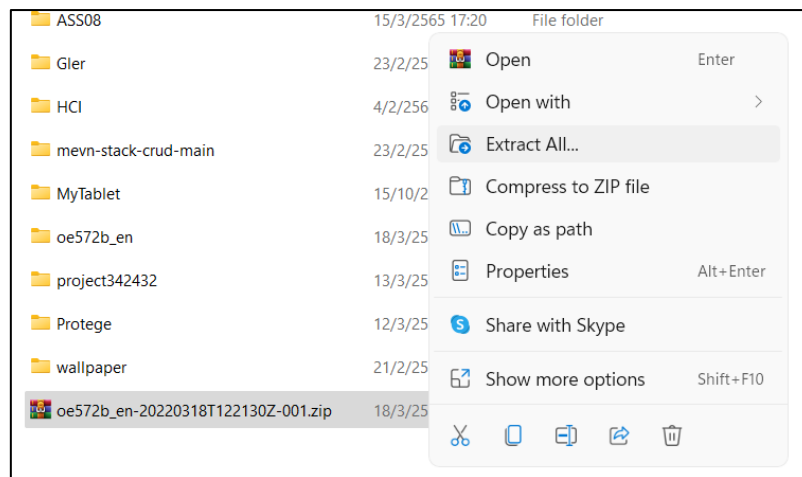
5.1.1. โหลดไฟล์โปรแกรม HOZO จาก

<https://drive.google.com/drive/folders/1pocWg4LbVLOYXcS-5pEV4tQlF-A-Ogs0b?usp=sharing> จากนั้น
จะได้ไฟล์ **oe572b_en-20220318T122130Z-001.zip**



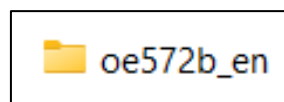
รูปที่ 33 ไฟล์โปรแกรม HOZO

5.1.2. ทำการแตกไฟล์ หรือ **Extract All** โดยสามารถทำการแตกไฟล์ไว้ที่ใดก็ได้ เนื่องจากเราไม่จำเป็นต้องติดตั้งโปรแกรม ดังนั้นหากต้องการถอนการติดตั้งโปรแกรม HOZO ก็สามารถลบไฟล์เดอร์ที่ Extract All ได้เลย



รูปที่ 34 แสดงวิธีการแตกไฟล์

5.1.3 จากนั้นจะได้โฟลเดอร์ **oe572b_en** ดังรูป ดับเบิลคลิกไปที่ไฟล์ จากนั้นคอมพิวเตอร์จะแสดงไฟล์ต่างๆภายในไฟล์ **oe572b_en** ดังรูป ดังนี้



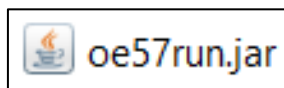
รูปที่ 35 โฟลเดอร์โปรแกรม HOZO

File Explorer view of the directory: This PC > New Volume (D:) > oe572b_en

Name	Date modified	Type	Size
data3	18/3/2565 19:23	File folder	
LocalStorage	18/3/2565 19:23	File folder	
manual	18/3/2565 19:23	File folder	
oe57run_lib	18/3/2565 19:23	File folder	
ServerStorage	18/3/2565 19:23	File folder	
temp	18/3/2565 19:23	File folder	
font.properties	18/3/2565 19:23	PROPERTIES File	1 KB
hozo.ini	18/3/2565 19:23	Configuration setti...	1 KB
hozo.properties	18/3/2565 19:23	PROPERTIES File	12 KB
hzprinter.properties	18/3/2565 19:23	PROPERTIES File	1 KB
icon.ico	18/3/2565 19:23	ICO File	57 KB
icon.jpg	18/3/2565 19:23	JPG File	13 KB
licence(hozo).txt	18/3/2565 19:23	Text Document	5 KB
locale	18/3/2565 19:23	File	1 KB
oe5.bat	18/3/2565 19:23	Windows Batch File	1 KB
oe5.command	18/3/2565 19:23	COMMAND File	1 KB
oe57run.jar	18/3/2565 19:23	Executable Jar File	6,706 KB
readme_jp.txt	18/3/2565 19:23	Text Document	3 KB
webdav.properties	18/3/2565 19:36	PROPERTIES File	1 KB

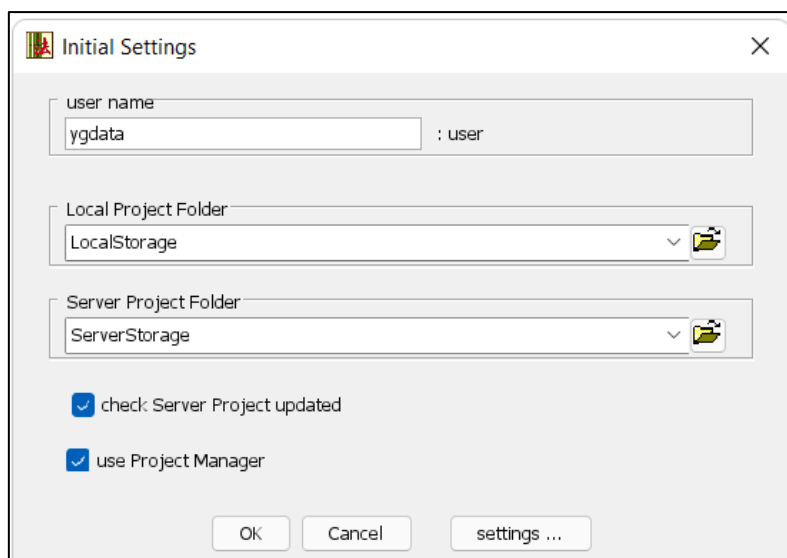
รูปที่ 36 ไฟล์ต่างๆภายในโฟลเดอร์ของโปรแกรม

5.1.4. จากนั้นทำการดับเบิลคลิกเข้าไปที่ไฟล์ **oe57run.jar** เพื่อทำการเรียกใช้งานโปรแกรม HOZO



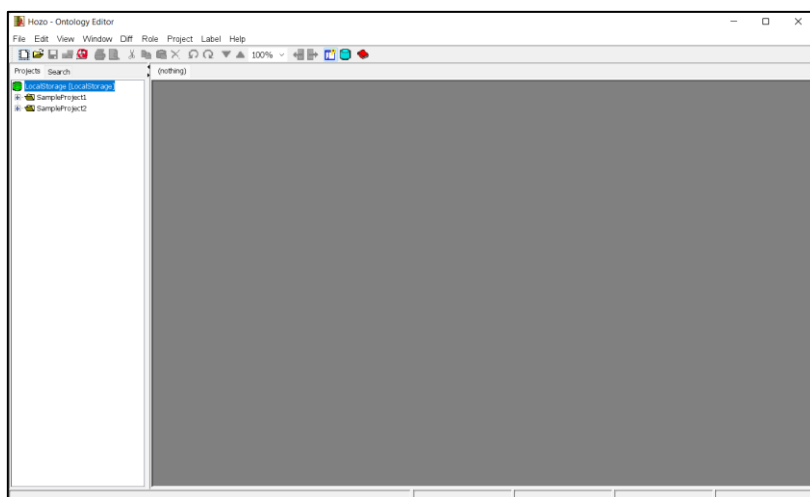
รูปที่ 37 ไฟล์จาวาที่ใช้รันโปรแกรม

5.1.5. จากนั้นโปรแกรม HOZO จะเปิดหน้าต่างนี้ขึ้น ในช่อง user name สามารถตั้งชื่ออื่นๆ ได้ ในคู่มือนี้จะตั้งตัวอย่าง user name เป็น **ygdata** หรือชื่ออื่นๆที่ต้องการ จากนั้นสามารถกดปุ่ม **OK** เพื่อเข้าใช้งานโปรแกรม



รูปที่ 38 ตั้งชื่อ user name

5.5.6. จากนั้นหากสำเร็จจะปรากฏดังรูป ดังนั้นสามารถใช้งานโปรแกรมได้

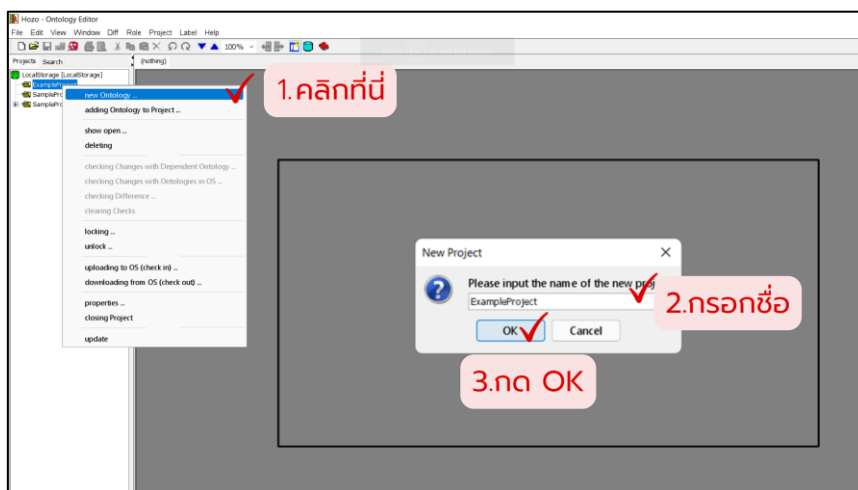


รูปที่ 39 ภาพเข้าใช้งานโปรแกรม HOZO

5.2.วิธีการสร้างโปรเจกต์

การสร้างโปรเจกต์ใหม่สามารถทำได้ 3 วิธีขึ้นอยู่กับความถนัดของผู้ใช้

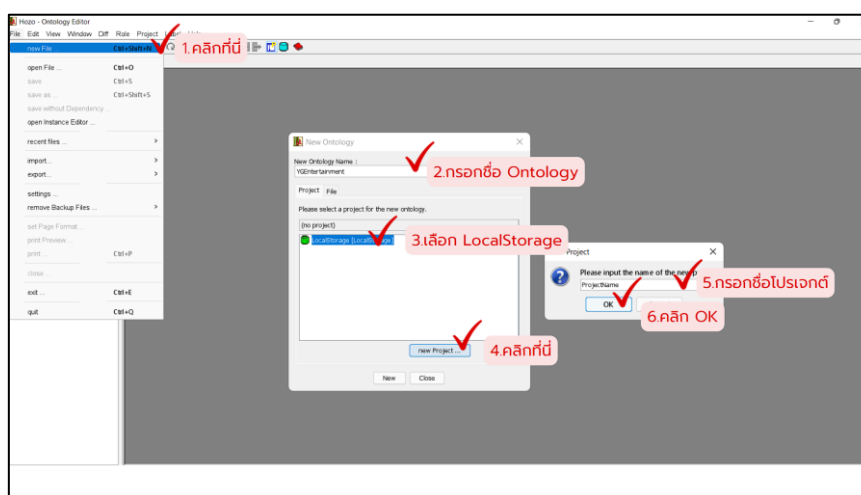
5.2.1 คลิกขวาที่ LocalStorage [LocalStorage]  -> new Project -> กรอกชื่อโปรเจกต์ -> กด OK เพื่อทำการสร้างโปรเจกต์



รูปที่ 40 วิธีการสร้างโปรเจกต์แบบที่ 1

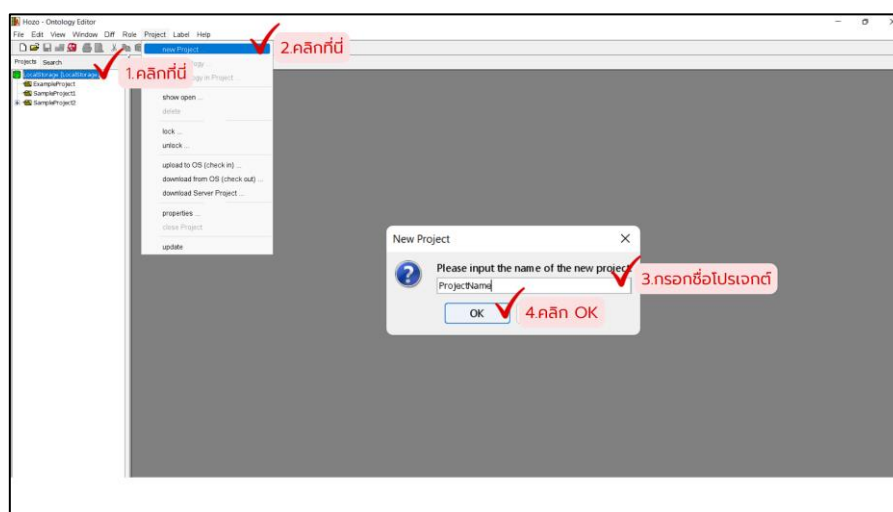
5.2.2. คลิก File -> new File -> กรอกชื่อ Ontology -> เลือก LocalStorage [LocalStorage] -> New Project -> กรอกชื่อโปรเจกต์

วิธีนี้จะสามารถสร้าง Ontology ไปพร้อมกันได้เลย



รูปที่ 41 วิธีการสร้างโปรเจกต์แบบที่ 2

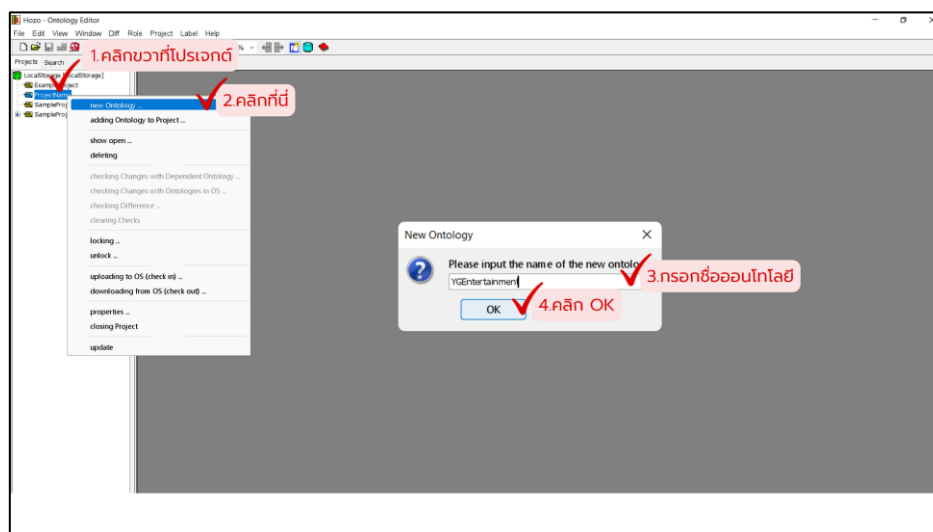
5.2.3. คลิก LocalStorage -> คลิกเมนู Project -> new Project -> กรอกชื่อโปรเจกต์ -> คลิก OK



รูปที่ 42 วิธีการสร้างโปรเจกต์แบบที่ 3

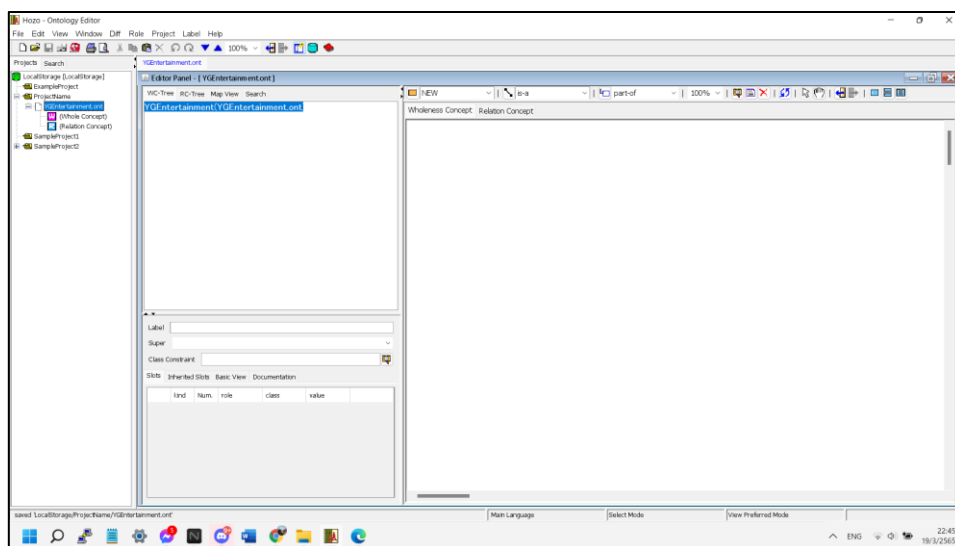
5.3.วิธีการสร้างออนโทโลยีใหม่ (New Ontology)

จากการสร้างโปรเจกต์ใหม่จะเห็นว่ามียังวิธีที่ 2 ที่มีการสร้างออนโทโลยีไปในขั้นตอนเดียวกันเลย ดังนั้นวิธีที่ 2 จึงไม่จำเป็นต้องสร้างออนโทโลยีใหม่อีกครั้ง มีเพียงวิธีที่ 1 และ 3 ที่ต้องดำเนินการต่อด้วยขั้นตอนนี้



รูปที่ 43 วิธีการสร้างออนโทโลยีใหม่

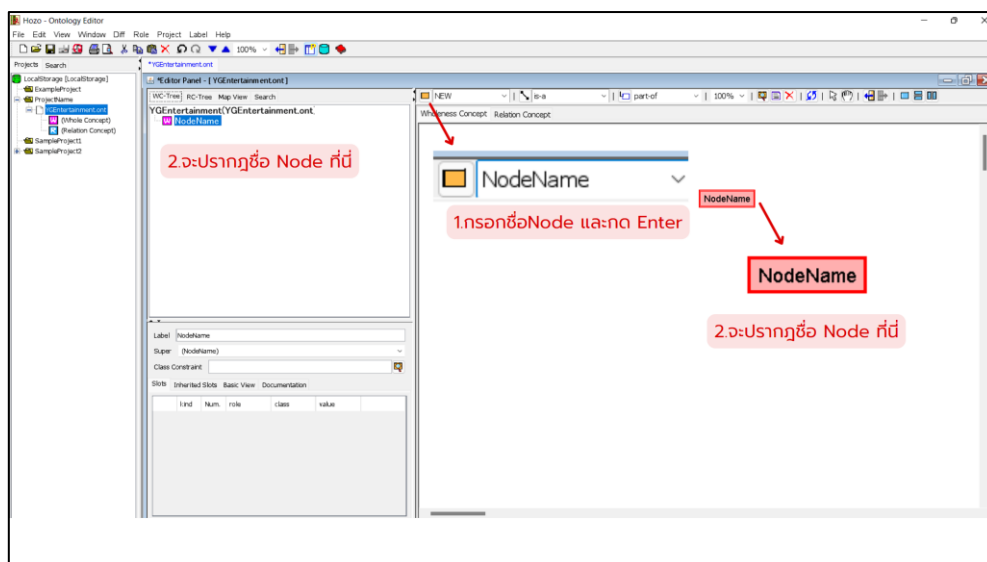
เมื่อสร้างเสร็จและดับเบิลคลิกไฟล์ออนโทโลยีที่เพิ่งสร้างจะได้ดังภาพ



รูปที่ 44 ภาพหน้าจอไฟล์ออนโทโลยี

5.4.วิธีการสร้างโหนดและซับโหนด (Add Node)

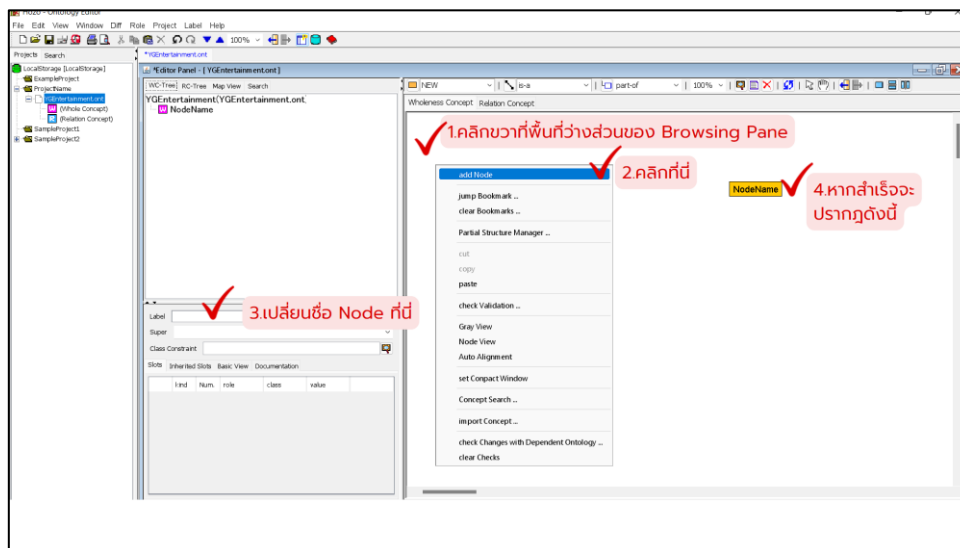
สามารถสร้างโหนด โดยการคลิกที่ไอคอนสร้างโหนด โดยจะสามารถสร้างได้ทีละ 1 โหนดโดยผู้ใช้งานสามารถเปลี่ยนชื่อโหนด ได้ตามต้องการ



รูปที่ 45 วิธีที่ 1 สำหรับการสร้างโหนด (Concept)

โดยอีกวิธีหนึ่งคือ คลิกขวาที่ส่วน Browsing Pane (ส่วนแสดงกราฟฟิก) จะปรากฏดังรูป จากนั้นคลิกตามขั้นตอน
ดังนี้

add Node -> เปลี่ยนชื่อ โหนด จาก New เป็นชื่อตามที่ต้องการ ผู้ใช้งานต้องการ ผู้ใช้สามารถเปลี่ยนชื่อโหนด (Concept)
ได้ที่ label ในส่วนของ Definition Pane

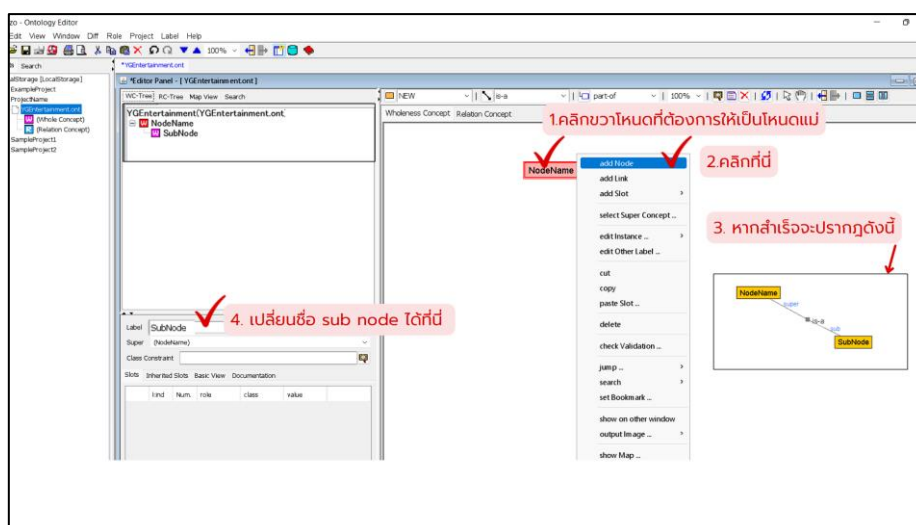


รูปที่ 46 วิธีที่ 2 สำหรับการสร้างโหนด (Concept)

หากผู้ใช้งานต้องการสร้างซับโหนด (Sub Node) สามารถทำได้ 3 วิธีโดยมีวิธีดำเนินการขั้นตอนดังนี้

5.4.1.เลือก Node ที่ต้องการให้เป็นโหนดแม่ -> คลิกขวา -> เลือก add Node

หากสำเร็จจะปรากฏดังรูป

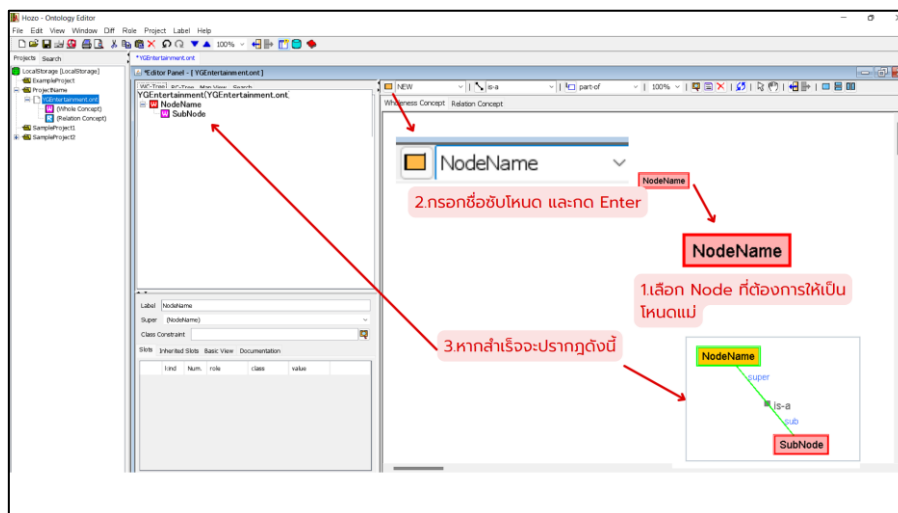


รูปที่ 47 วิธีที่ 1 สำหรับการสร้างซับโหนด

5.4.2.เลือก Node ที่ต้องการให้เป็นโนดแม่ -> คลิกที่ไอคอนสร้าง Node ในพื้นที่ Editing Tool Bar -

> พิมพ์ชื่อ SubNode ที่ต้องการ

หากสำเร็จจะปรากฏดังรูป

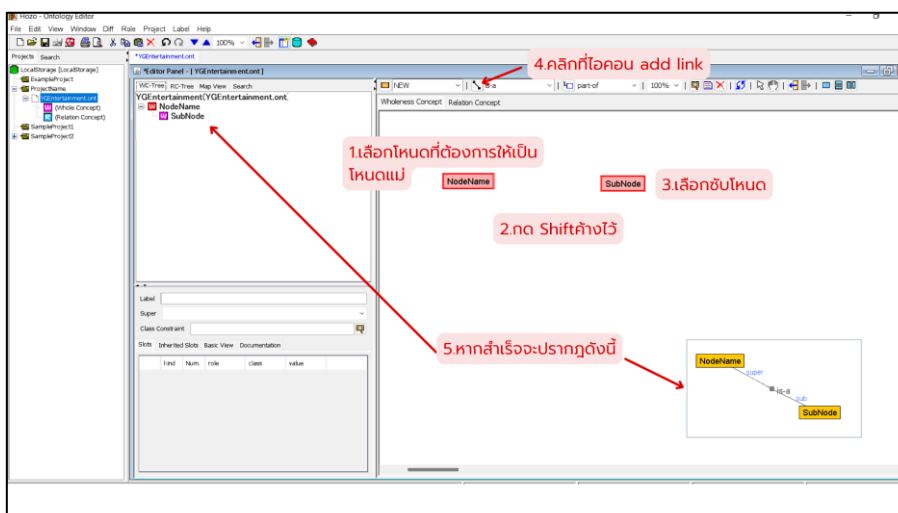


รูปที่ 48 วิธีที่ 2 สำหรับการสร้างซบโนด

5.4.3.วิธีนี้เป็นวิธีสร้างความสัมพันธ์ระหว่างสองโนดที่ไม่มีความสัมพันธ์กันมาก่อน

เลือก Node ที่ต้องการให้เป็นโนดแม่ -> กดปุ่ม Shift บนแป้นพิมพ์ค้างไว้ -> เลือก Node ที่ต้องการให้เป็น Sub Node -> คลิกที่ไอคอน add link เพื่อสร้างความสัมพันธ์ระหว่างโนด

หากสำเร็จจะปรากฏดังรูป



รูปที่ 49 วิธีที่ 3 สำหรับการสร้างซบโนด

5.5.วิธีการเพิ่ม Slot (Add Slot)

เป็นการกำหนดคุณสมบัติของมโนทัศน์ (Property) หรือเป็นการกำหนดส่วนประกอบหรือคุณสมบัติของ Node นั้นๆ โดยประเภทความสัมพันธ์ มี 2 ประเภทดังนี้

1. part-of เป็นการกำหนดส่วนประกอบ
2. attribute-of เป็นการกำหนดคุณลักษณะ

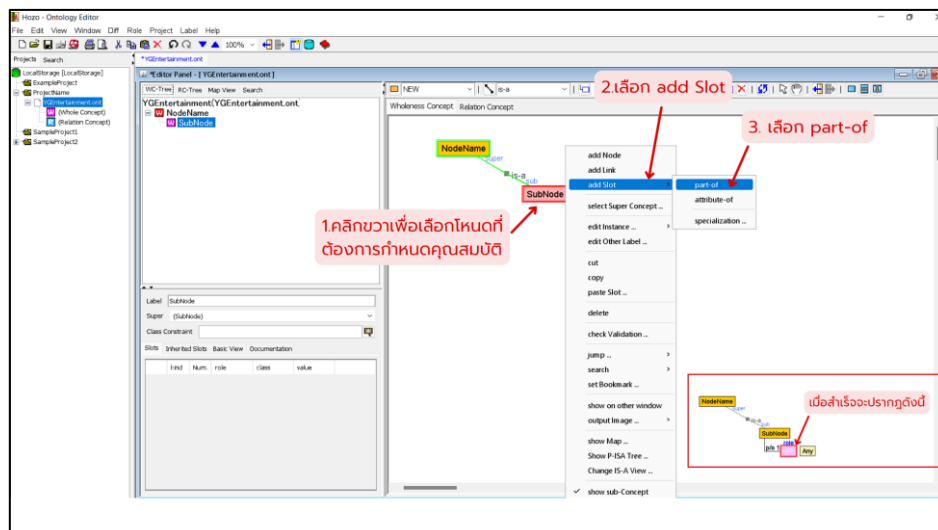
โดยจะยกตัวอย่างออนโทโลยีดอกไม้ เช่น

3. part-of ได้แก่ ดอก, ใบ, กิ่ง, ก้าน, ลำต้น, ราก เป็นต้น
4. attribute-of ได้แก่ สายพันธุ์, สีของดอก เป็นต้น

ในการเพิ่มคุณสมบัติภายในโปรแกรมมีขั้นตอนดังนี้

5.5.1. การเพิ่มคุณสมบัติแบบ part-of

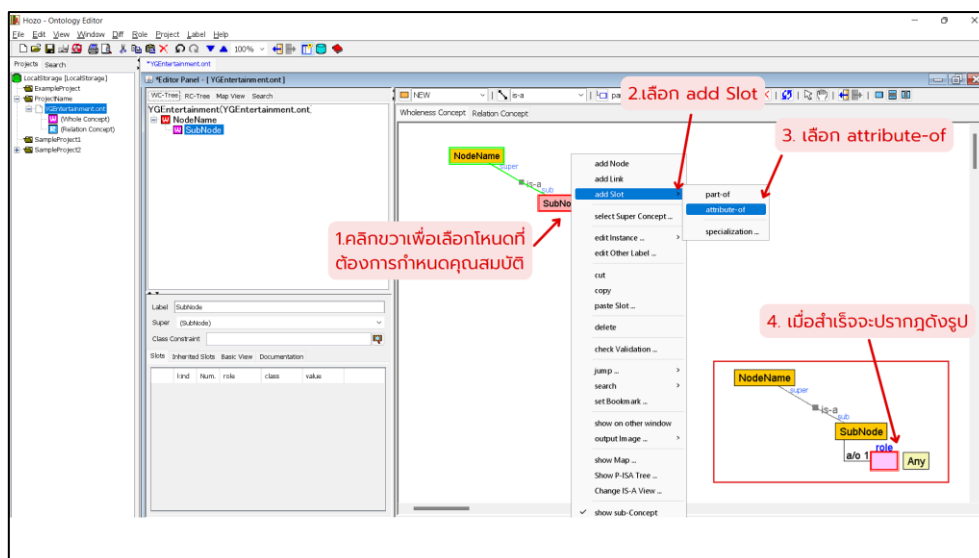
คลิกขวาเพื่อเลือกโหนดที่ต้องการกำหนดคุณสมบัติ -> เลือก add Slot -> เลือก part-of



รูปที่ 50 วิธีเพิ่มคุณสมบัติแบบ part-of

5.5.2. การเพิ่มคุณสมบัติแบบ attribute-of

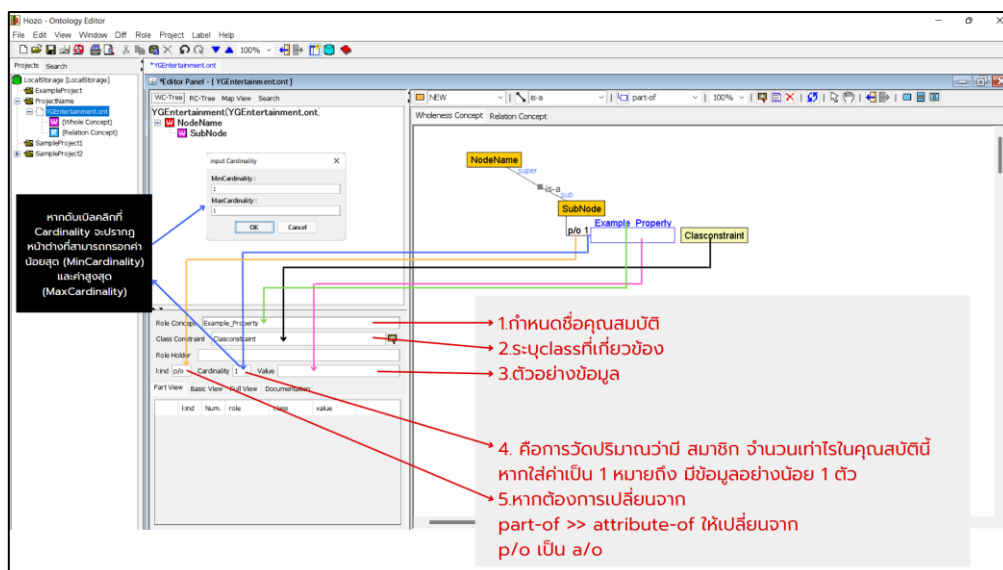
คลิกขวาเพื่อเลือกโหนดที่ต้องการกำหนดคุณสมบัติ -> เลือก add Slot -> เลือก attribute -of



รูปที่ 51 วิธีเพิ่มคุณสมบัติแบบ attribute -of

จากนั้นผู้ใช้งานสามารถกำหนดรายละเอียดขององค์ประกอบเบื้องต้นได้ดังนี้

6. Role Concept - กำหนดชื่อคุณสมบัติ
7. Class Constraint - ระบุคลาสที่เกี่ยวข้อง
8. Value - ตัวอย่างข้อมูล
9. Cardinality - คือการวัดปริมาณว่ามี สมาชิก จำนวนเท่าไรในคุณสมบัตินี้ หากใส่ค่าเป็น 1 หมายถึง มีข้อมูลอย่างน้อย 1 ตัว
10. Kind - หากต้องการเปลี่ยนจาก part-of -> attribute-of ให้เปลี่ยนจาก p/o เป็น a/o

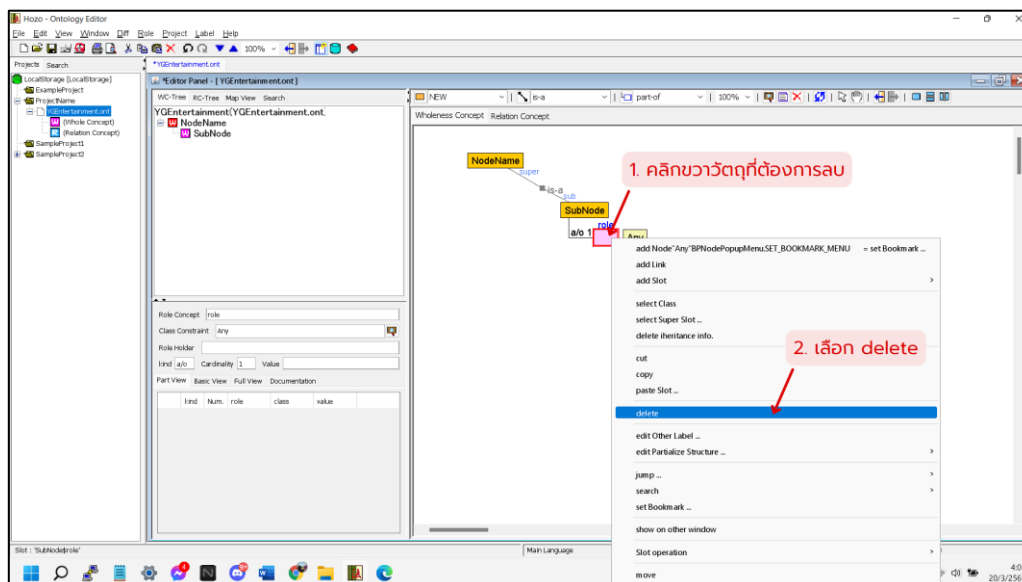


รูปที่ 52 วิธีกำหนดคุณสมบัติแบบ part-of/attribute-of

5.6.วิธีการลบวัตถุ (Deleting Object)

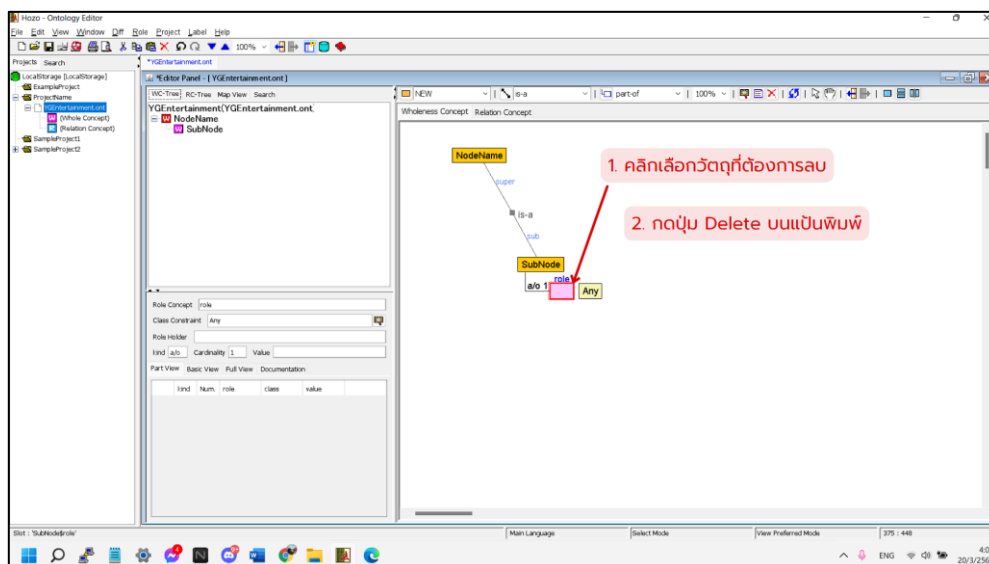
ผู้ใช้งานสามารถลบวัตถุได้ 2 วิธี คือ

5.6.1. คลิกขวาที่วัตถุ ที่ต้องการลบ -> เลือก Delete



รูปที่ 53 วิธีที่1 สำหรับการลบวัตถุ

5.6.2. คลิกเลือกวัตถุ ที่ต้องการลบ -> กดปุ่ม Delete บนแป้นพิมพ์

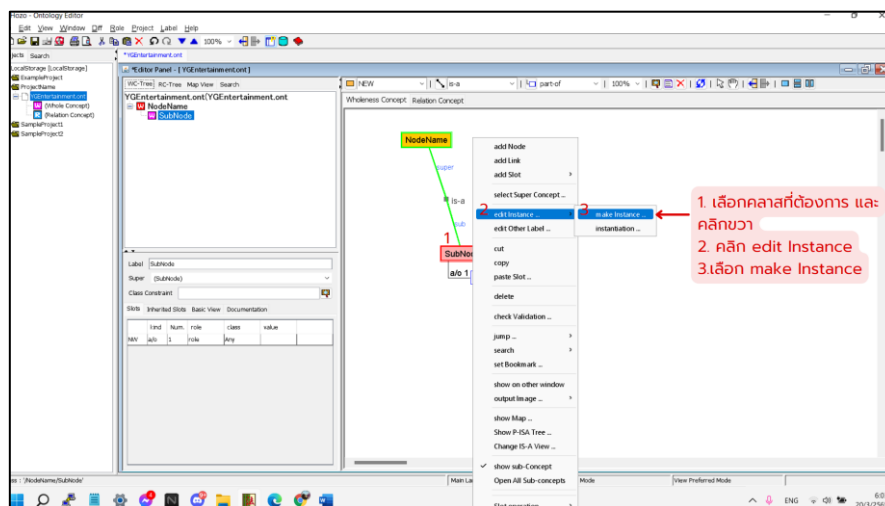


รูปที่ 54 วิธีที่2 สำหรับการลบวัตถุ

5.7.วิธีการสร้าง Instance

เป็นการสร้างข้อมูลตัวอย่าง หลังจากที่มีการสร้าง Schema Level แล้ว เราก็ควรจะมีการแสดงข้อมูลตัวอย่างหรือ Instance Level เพื่อให้เข้าใจใน Ontology ที่สร้างมากยิ่งขึ้น วิธีการสร้าง Instance มีดังนี้

คลิกขวาคลาสที่ต้องการ -> เลือก edit Instance -> เลือก Make Instance

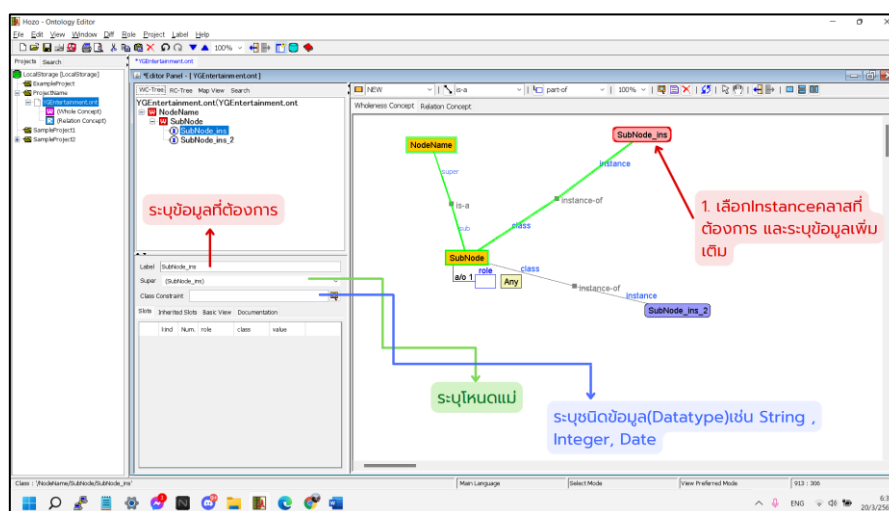


รูปที่ 55 วิธีสร้าง Instance

จากนั้นจะแสดงดังรูป และทำการระบุข้อมูลของ Instance class ที่เพิ่มเข้ามาโดย

เลือก Instance class ที่ต้องการ และระบุข้อมูลเพิ่มเติม -> ระบุ label -> ระบุ -> Super Class -> ระบุ Class

Constraint

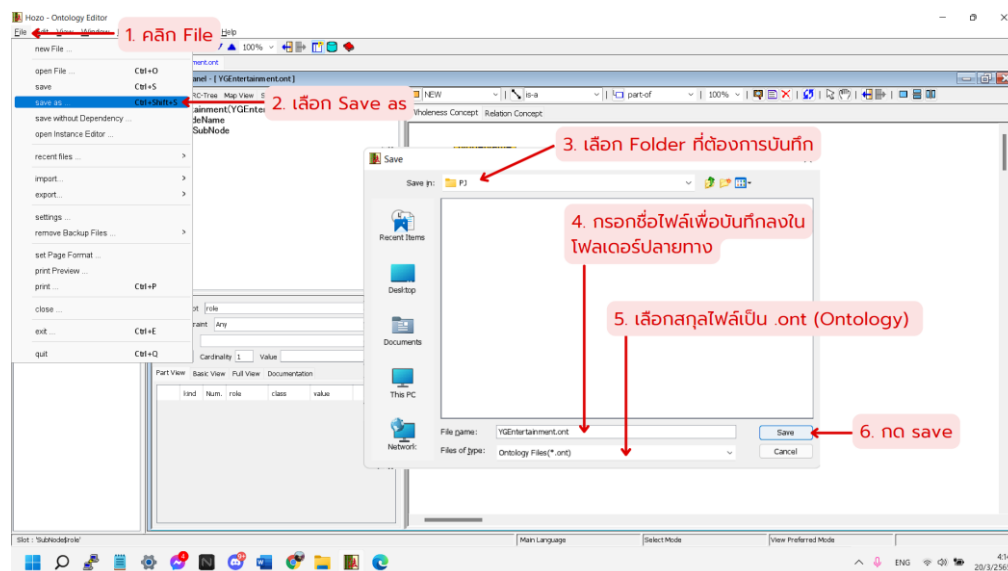


รูปที่ 56 วิธีสร้าง Instance

5.8.วิธีการบันทึกไฟล์ (Save File)

หากเป็นการบันทึกไฟล์ครั้งแรก ผู้ใช้งานสามารถบันทึกไฟล์ตามขั้นตอนดังนี้

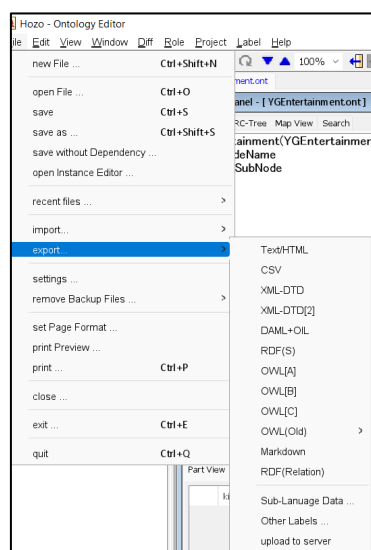
คลิก File -> เลือก Save as -> เลือก Folder ที่ต้องการบันทึก -> กรอกชื่อไฟล์เพื่อบันทึกลงในโฟลเดอร์ปลายทาง
-> เลือกสกุลไฟล์เป็น .ont (Ontology) -> กด save



รูปที่ 57 วิธีการบันทึกไฟล์

5.9.วิธีการ Export ไฟล์

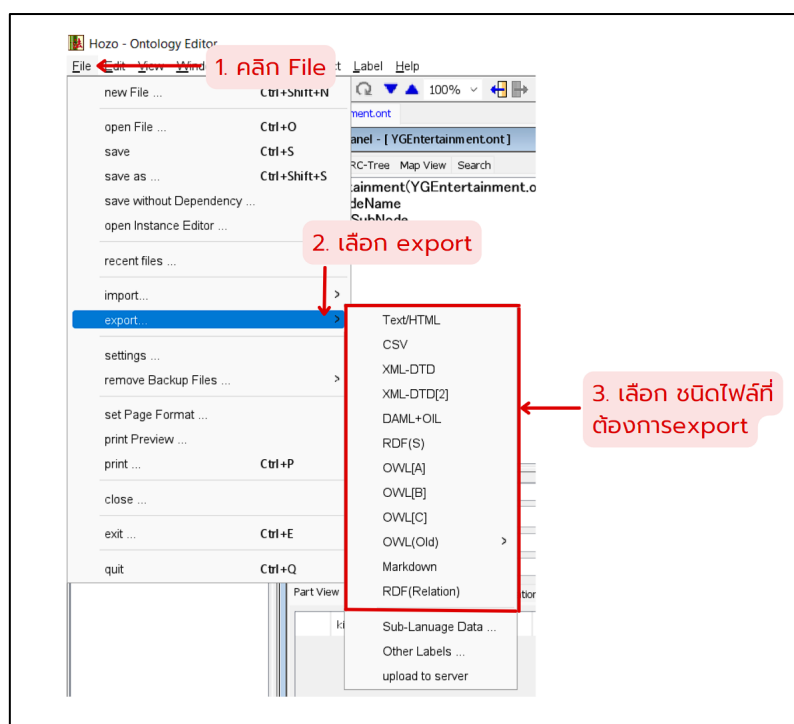
ผู้ใช้งานสามารถ export ไฟล์ออนโทโลยีเพื่อสามารถนำไปใช้งานได้สะดวกขึ้น ซึ่งโปรแกรมสามารถให้ผู้ใช้งาน export ไฟล์ได้หลายชนิด ดังนี้ Text/HTML, 2CSV, XML-DTD, XML-DTD[2], DAML+OIL, RDF(S), OWL[A], OWL[B], OWL[C], OWL(Old), Markdown, RDF(Relation)



รูปที่ 58 ตัวอย่างชนิดไฟล์ของการ export

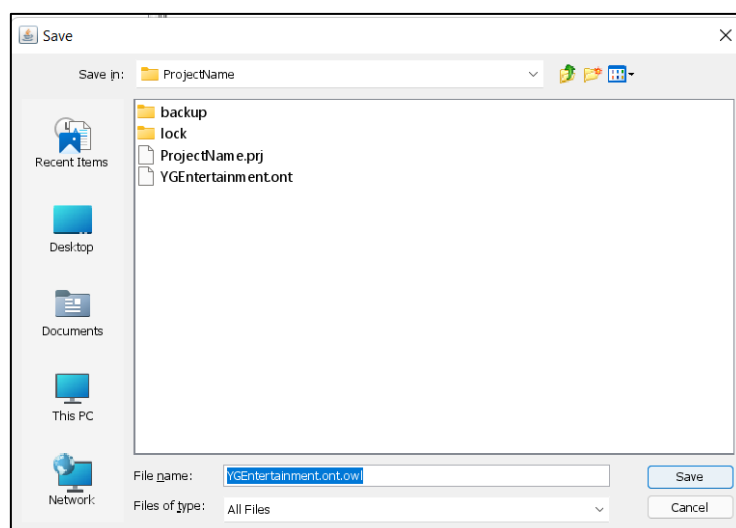
ผู้ใช้งานสามารถ export ไฟล์ออนโทโลยีตามขั้นตอน ดังนี้

คลิก File -> Export -> ชนิดไฟล์ที่ต้องการ Export



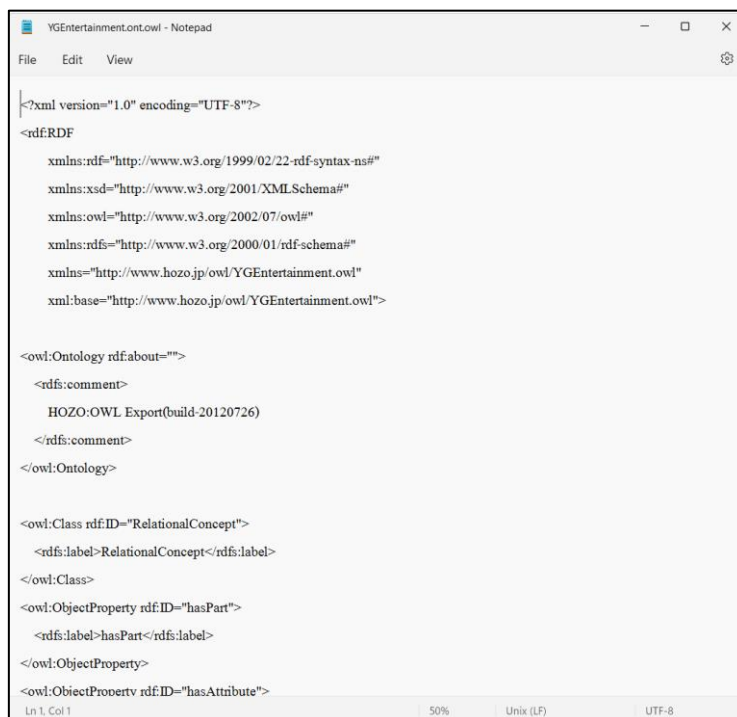
รูปที่ 59 วิธีการ export ไฟล์

จากนั้นโปรแกรมจะแสดงหน้าต่างหนึ่ง หลักจากเลือกรูปแบบชนิดไฟล์ที่ต้องการ export ตัวอย่างเช่น
ทำการ export file ในรูปแบบไฟล์ OWL[A] จะปรากฏดังรูป จากนั้นเลือกที่อยู่ของไฟล์เพื่อทำการบันทึกไว้ในโฟลเดอร์ที่ต้องการ



รูปที่ 60 เลือกที่อยู่ไฟล์เพื่อบันทึก

จากนั้นทำการทดลองเปิดไฟล์ OWL ในโปรแกรม NotePad ก็พบว่า จากที่ทำในโปรแกรมจะเป็นเพียงการสร้าง class ไม่จำเป็นต้องมีการเขียนโค้ด ทำให้เกิดความผิดพลาดน้อยและง่ายต่อการเข้าใจ เนื่องจากรูปแบบในการมองเห็นอยู่ในรูปแบบของ Class และ Subclass เมื่อผู้ใช้งานทำการ Export file ในรูปแบบของไฟล์อื่นๆ ข้อมูลภายในโปรแกรม HOZO ก็จะถูกแปลงเป็นวิธีเขียนของภาษานั้นๆ



```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.hozo.jp/owl/YGEntertainment.owl"
  xml:base="http://www.hozo.jp/owl/YGEntertainment.owl">

  <owl:Ontology rdf:about="">
    <rdfs:comment>
      HOZO:OWL Export(build-20120726)
    </rdfs:comment>
  </owl:Ontology>

  <owl:Class rdf:ID="RelationalConcept">
    <rdfs:label>RelationalConcept</rdfs:label>
  </owl:Class>

  <owl:ObjectProperty rdf:ID="hasPart">
    <rdfs:label>hasPart</rdfs:label>
  </owl:ObjectProperty>

  <owl:ObjectProperty rdf:ID="hasAttribute">

```

รูปที่ 61 ตัวอย่างข้อมูลไฟล์ OWL[A]

5. ความต้องการทางด้าน Hardware และ Software ของเครื่องมือ

ตารางที่1 ความต้องการทางด้าน Hardware และ Software ของเครื่องมือ

CPU	Intel Pentium III Processor, 800MHz หรือสูงกว่า
Ram	256MB หรือสูงกว่า
Hard Disk	30MB หรือมากกว่า
ความละเอียดของ Monitor	1024x768 (XGA) หรือสูงกว่า
OS	Windows 2000 (แนะนำ Windows XP หรือสูงกว่า), Mac OS X
JAVA	JRE/JDK 1.5.0_10 หรือเวอร์ชันสูงกว่า (แนะนำ 1.6.0 หรือสูงกว่า)
Network	ไม่จำเป็นต้องใช้เครื่องท้องถิ่น (Local) กรณีมีการใช้งานร่วมกันภายในองค์กร เครือข่าย ลูกข่ายและเซิร์ฟเวอร์ให้เชื่อมต่อผ่าน Ethernet โดยใช้ TCP/IP Protocol

อ้างอิง

- [1] ห้องปฏิบัติการวิจัยเทคโนโลยีภาษาธรรมชาติและความหมายศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. (2556). คู่มือการใช้งานHozo-Ontology Editor. ค้นเมื่อ 18 มีนาคม 2565, จาก https://lst.nectec.or.th/oam_en/document_doc/Hozo_ThaiManual_25550123.pdf