enGeno Library

Properties Details

Name, Value Type	Description
Diagram *{go.diagram}*	Get หรือ Set diagram object และ template ของแผนผังครอบครัว
contextNode *{go.contextMenu}*	Get หรือ Set context menu object และ template ของ context menu ที่จะแสดงให้เห็นเมื่อคลิกขวาที่ตัวโหนด
originalArray *{array}*	Get หรือ Set ค่า Data array หลัก ที่ใช้สร้างแผนผังครอบครัว

Method Details

Name, Value Type	Description
enGeno(data,divId) *{diagram}*	ฟังก์ชั่น Create object enGeno **Parameters :** *{objectArray}***data** ข้อมูลที่ผู้ใช้ต้องการ initial ให้ diagram *{string}***divId** เป็น string ระบุชื่อ Div (HTML segment) ซึ่งเป็นเป้าหมายในการวาด diagram **Returns :** *{ diagram }*ที่ถูก set ค่าแล้ว
addChild(node,gender,data) *{int}*	ฟังก์ชันที่จะสร้างโหนดใหม่ขึ้นมา และตั้งค่า Data node ตามพารามิเตอร์ที่ได้รับ และทำการใส่โหนดใหม่ลงในแผนผังครอบครัว **Parameters :** *{object}***node** Object ที่ระบุโหนดเป้าหมายที่จะทำการเพิ่มบุตร *{string}***gender** เป็น string ระบุเพศของโหนดลูก เช่น “F”, “M”, “I” *{object}***data** เป็น object ของ data ที่จะใช้เป็นข้อมูลของโหนดลูก **Returns :** *{int}*ที่ระบุ key ของโหนดลูก

addDaughter(node,data) *{int}*	จะเรียกใช้ฟังก์ชัน addChild(node,gender,data) โดยระบุ gender เป็น “F” **Parameters :** *{object}***node** Object ที่ระบุโหนดเป้าหมายที่จะทำการเพิ่มบุตร *{object}***data** เป็น object ของ data ที่จะใช้เป็นข้อมูลของโหนดลูก **Returns :** *{int}*ที่ระบุ key ของโหนดลูก
addSon(node,data) *{int}*	จะเรียกใช้ฟังก์ชัน addChild(node,gender,data) โดยระบุ gender เป็น “M” **Parameters :** *{object}***node** Object ที่ระบุโหนดเป้าหมายที่จะทำการเพิ่มบุตร *{object}***data** เป็น object ของ data ที่จะใช้เป็นข้อมูลของโหนดลูก **Returns :** *{int}*ที่ระบุ key ของโหนดลูก
addSpouse(node,data) *{int}*	ฟังก์ชันจะสร้างโหนดใหม่โดยมี Default เป็นเพศที่ตรงข้ามกับโหนดเป้าหมาย ก็ต่อเมื่อโหนดที่เป็นเป้าหมายต้องยังไม่มีคู่สมรส โดยเช็คได้จากการเรียกฟังก์ชัน findMarriageArray(node) และค่าที่ return มาเป็น empty array **Parameters :** *{object}***node** Object ที่ระบุโหนดเป้าหมายที่จะทำการเพิ่มบุตร *{object}***data** เป็น object ของ data ที่จะใช้เป็นข้อมูลของโหนดลูก **Returns :** *{int}*ที่ระบุ key ของโหนดลูก
addParents(node) *{int}*	ฟังก์ชันจะทำการสร้างโหนดพ่อ และโหนดแม่ และเซ็ทข้อมูล attribute ‘m’ และ ‘f’ ในโหนดเป้าหมายให้สอดคล้องกันโหนดพ่อแม่ **Parameters :** *{object}***node** Object ที่ระบุโหนดเป้าหมายที่จะทำการเพิ่มโหนดพ่อแม่

copyJSON(obj) *{int}*	นำพารามิเตอร์ที่เข้ามาแปลงเป็น string จากคำสั่ง JSON.stringify() และแปลงกับมาเป็นค่าของ JSON object ด้วยคำสั่ง JSON.prase() เพื่อเป็นการทำ copy by value **Parameters :** *{object||objectArray}***object** Object ที่จะทำการ copy ค่า **Returns :** *{object||objectArray}*JSON object ที่ copy มาจาก parameter
editNodeAttribute(node,attr,value)	เปลี่ยน attribute ในโหนดที่ใส่เข้ามาเป็นพารามิเตอร์ **Parameters :** *{object}***node** Object ที่จะทำการเปลี่ยนค่า *{string}***attr** ระบุชื่อ attribute ที่ต้องการ set ค่า *{value}***value** ค่าที่จะนำมาใส่ attribute นั้นๆ
export() *{textFile}*	สร้าง text file จากข้อมูลในแผนผังครอบครัวตาม format ที่สามารถ readFile ได้เพื่อรีเทรินให้ผู้ใช้ **Returns :** *{textFile}*จากข้อมูล text file ที่ได้สร้างไว้
findParentsMarriageLabelNode(node) *{go.Link}*	ค้นหา Link ของการแต่งงาน ของโหนดที่ได้รับมาเป็นพารามิเตอร์ **Parameters :** *{object}***node** Object เป้าหมาย **Returns :** *{go.Link||null}*เส้นความสัมพันธ์ที่ระบุการแต่งานถ้าไม่เจอ คืนค่า null
genKey() *{int}*	วน loop ค้นหาคีย์ที่สามาถรใช้ได้ แบบไม่ซ้ำ และมีค่าน้อยที่สุด **Returns :** *{int}*คีย์ที่สามารถใช้งานได้
getCurrentDataNode() *{objectArray}*	นำ Data ของ diagram ในปัจจุบันออกมา และแปลงข้อมูลให้อยู่ใน Format ที่ผู้ใช้สามารถนำไปใช้งานต่อได้ **Returns :** *{objectArray}*Data จาก diagram ในปัจจุบัน
getOriginalArray() *{objectArray}*	คืนค่าตัวแปร originalArray **Returns :**

	*{objectArray}*จาก ตัวแปร originalArray
getSelectedNodes() *{objectArray}*	ฟังก์ชันเช็คโหนดทั้งหมดว่ามีโหนดไหนมีค่า isSelected เป็น ‘true’ และ สร้างตัวแปรอาเรย์ขึ้นมาเพื่อเก็บโหนดนั้นๆ **Returns :** *{objectArray}*โหนดทั้งหมดที่มีค่า isSelected เป็น true
hasKey(num) *{boolean}*	ตรวจสอบว่า มีคีย์ที่ส่งเข้ามาเป็นพารามิเตอร์นั้น อยู่ใน Data array ของ diagram หรือยัง **Parameters :** *{int}***num** คีย์ที่จะตรวจสอบ **Returns :** *{boolean}*ถ้ามีคีย์นั้นๆแล้วคืนค่า ‘true’ ยังไม่มีคืนค่า ‘false’
init()	คำสั่งที่ทำให้มีการ setup Parents และ setup Marriage ใช้ในตอนเริ่มต้นเท่านั้น
makeImage(para1, para2) *{image}*	ฟังก์ชันสร้าง image ที่ extend มาจาก ‘makeImage()’ ใน GoJs เพื่อสร้างไฟล์รูปภาพ โดยแบ่งฟังก์ชันออกเป็น 4 รูปแบบ รูปแบบที่ 1 ใส่ para1 อย่างเดียว : จะเป็นการระบุ Scale รูปแบบที่ 2 ใส่ para1 และ para2 : จะเป็นการระบุความกว้าง และยาว รูปแบบที่ 3 ใส่ para1 เป็น object :เป็นการระบุ propoty ของ image file เอง รูปแบบที่ 4 ไม่ใส่ parameter : จะเป็นค่า default ที่ฟังก์ชันกำหนดไว้ให้ **Parameters :** *{int || object}***para1** ถ้าเป็น int แล้วไม่มี para2 จะเป็นการกำหนวดค่า scale ถ้ามี para2 จะเป็นการกำหนอกค่า width ถ้ามี type เป็น object จะต้องไม่มี para2 *{int}***para2** กำหนดค่า height ของรูปภาพ **Returns :** *{image}*รูปภาพที่ถูกสร้างด้วยฟังก์ชัน makeImage() ใน GoJs
pushObjInOriginalArray(obj)	เพิ่ม obj เข้าไปในตัวแปร originalArray **Parameters :** *{object}***obj** ข้อมูลโหนดที่ต้องการเพิ่มใน originalArray

readFile(event) *{objectArray}*	อ่านไฟล์ที่รับมากจากการเรียก Browse file จาก input type ‘file’ ใน HTML segment **Parameters :** *{event}***event** ที่ระบุ target file **Returns :** *{objectArray}*ผลลัพธ์ที่ได้จากการอ่านไฟล์
removeNode(node)	ลบโหนดที่ระบุมาเป็นพารามิเตอร์ และข้อมูลของโหนดอื่นที่เกี่ยวข้องกับโหนดที่ระบุ **Parameters :** *{object}***node** Object เป้าหมาย
removeNullObject(arr)	ลบค่าช่องที่เป็น null ในอาเรย์ที่ระบุมาเป็นพารามิเตอร์ **Parameters :** *{array}***arr** array เป้าหมาย
resetDiagram(arr)	ทำการสร้าง diagram ขึ้นมาใหม่ โดยเซ็ทตัวแปร originalArray ให้สอดคล้องกับ arr ที่ใส่เข้ามา และทำการ setup diagram ใหม่โดยใช้ข้อมูลจาก originalArray **Parameters :** *{objectArray }***arr** ข้อมูลที่จะนำมา generate diagram ใหม่
resetOriginalArray()	Set ค่าตัวแปร originalArray ให้สอดคล้องกับสถานะ data ใน diagram ปัจจุบัน โดยการอ้างอิงข้อมูลจากการเรียกฟังก์ชัน getCurrentDataNode()
searchByKeyword(keyword) *{objectArray}*	ฟังก์ชันที่หา string matching ระหว่าง keyword กับ attribute ‘comment’ ใน DataArray และหาว่า keyword ที่ส่งเข้ามีมีความสอดคล้องกับ attribute ‘a’ในแต่ละโหนดหรือเปล่า หรือเปล่า **Returns :** *{objectArray}*ผลลัพธ์จากการค้นหา
setContextNode(contextMunu)	ฟังก์ชันสำหรับการ set context menu ที่เกิดจากการคลิกขวาในแต่ละโหนด **Parameters :** *{ go.Adornment }***context** รูปแบบของ Context menu ที่ใช้ GoJs’s elements ในการสร้าง
setOriginalArray(data)	เซ็ทค่าตัวแปร originalArray **Parameters :** *{objectArray }***data** ข้อมูลใหม่ที่จะใช้สร้าง diagram