جولة حول العقد الأساسية

مقدمة :

في هذا الدرس سوف نلقي نظرة على مجموعة أساسية من العقد الموجودة افتراضيا في Node-REDثم عرض مجموعة موسعة من العقد التي يدعمها FRED .

لكل عقدة، سترى لمحة موجزة عن وظائفها، و إشارة منها إلى مثال في سلسلة الدروس ، بحيث يمكنك النظر بمزيد من التفاصيل في كيفية التعامل مع العقدة، هذا الدرس هو قسم مرجعي. ومع ذلك، يجب أن يكون لديك فكرة عن الوظائف الأساسية المتاحة لك خلال استخدام العقد عندما تبدأ في صياغة التدفقات الخاصة بك .

عقد الـ Node-RED الإفتراضية :

عندما يتم تثبيت Node-RED على جهازك مثل الراسبيري باي و Beagleboard ، ستظهر لك المجموعة الافتراضية من العقد. هناك ثمانية فئات رئيسية من العقد في التثبيت الافتراضي : المدخلات(input) ، المخرجات (functions ، (output ، والاجتماعي (social) و التخزين (storage)و تحليل (analysis) ، المتقدمة (advanced) و الراسبيري باي ، قم بإلقاء نظرة على الشكل التالي :

عقد الإدخال (Input Node) :

هناك 7 عقد المدخلات الأساسية المثبتة بشكل افتراضي . أنها تغطي اليات الاتصال الأساسية التي يتم استخدامها في تطبيقات انترنت الأشياء . بدأ من البروتوكولات منخفضة المستوى مثل UDP و TCP من خلال البروتوكول عالي المستوى HTTP و النشر/والاشتراك MQTT .

اسم العقدة	الوصف	أمثلة على العقد
inject	لتوليد الطابع الزمني أو نص تم تكوينه من قبل المستخدم في رسالة . يمكن تكوينها يدوينا بإستخدام زر الضخ على الجانب الأيسر للعقدة أو في أوقات محددة باستخدام (Cron) .	بناء التدفق الأول (Hello World)
catch	تقوم بإلتقاط الأخطاء التي ألقاها العقد الأخرى على نفس التبويب. إذا ألقت عقدة خطأ أثناء التعامل مع الرسالة، فإن التدفق في العادة يتوقف. هذة العقدة يمكن استخدامها لالتقاط تلك الأخطاء و إرجاع الرسالة مع تفصيل الخطأ و عقدة المصدر و النوع.	
mqtt	للإشتراك إلى MQTT broker و الاستماع على موضوع معين، فتعيد أي بيانات منشورة حول هذا الموضوع كرسالة جديدة . ويدعم مستويات الخدمة و الاحتفاظ بالبيانات الأخيرة .	تلقي JSON عبر رسالة MQTT
http	يتلقى طلبات HTTP، مما يسمح للـ Node-RED لتكون بمثابة خادم الويب الأساسي. يتم تسليم هيكل HTTP كرسالة مخرجة كرد. يمكن أن تحتوي الرسالة على URL – البيانات المشفرة أو JSON .	بناء خدمة ويب بسيطة باستخدام عقد HTTP

websocket	يوفر نقطة النهاية للمتصفح لتأسيس اتصال websocket مع Node-RED . يقدم اتصال duplex لمجموعات المتصفح/الخادم .	استخدام Websockets مع Node-RED
tcp	تستخدم لقبول طلبات TCP الواردة على منفذ محدد أو للاتصال إلى منفذ TCP البعيد . تنتج رسائل تحتوي على بيانات TCP حيث تولد رسائل تحتوي على بيانات TCP إما منفردة أو سلسلة من buffer أو string أو base64 encode .	
udp	تستخدم لقبول حزم(مجموعة الرسائل)UDP الواردة على منفذ معين. يولد الرسائل التي تحتوي على بيانات UDP كـ , BUFFER String أو base64 string .	
serial in	يقرأ من المنفذ التسلسلي على الجهاز المحلي. يمكن إعدادها لقراءة buffers أو لفترة زمنية محدودة أو عند الأنتهاء من السطر.	

عقد الإخراج (Output Nodes) :

عقد الإخراج(output Nodes) هي صورة طبق الأصل من المجموعة الأساسية لعقد الإدخال (input Nodes) و توفر وسيلة لإرسال البيانات على نفس مجموعة البروتوكولات mqtt,http,UDP .

اسم العقدة	الوصف	أمثلة على العقد
debug	يوفر وسيلة بسيطة لعرض الرسائل على لوحة الإخراج (تبويب debug) . يمكن تهيئتها لعرض msg.payload فقط او msg object كامل .	بناء التدفق الأول (Hello World) تنبيهات الطقس تلقي JSON عبر رسالة MQTT استخدام عقدة الإخراج MQTT لإختبار التدفق
mqtt	للإشتراك في MQTT broker و مشاركة اي بيانات (msg.payload) التي تتلقاها في الرسائل الواردة إلى الموضوع. وهو يعدم جودة مستويات الخدمة و الاحتفاظ في البيانات الأخيرة.	استخدام عقدة الإخراج MQTT لإختبار التدفق
http	ترسل إجابات إلى طلبات HTTP الواردة من عقدة إدخال HTTP . ويتم تحديد محتوى الرسالة من خلال msg.payload ، و يمكن أن تحتوي على رأس وتعريف رمز الحالة .	بناء خدمة ويب بسيطة باستخدام عقد HTTP
websocket	يرسل msg.payload على websocket . إذا تم تعريب msg_session ، فإنه يرسل إلى العميل الأصلي، وإلا يتم بثه إلى جميع العملاء.	
tcp	الردود على منفذ TCP الذي تم إعداده. ويمكن أيضا أن تستخدم للإرسال إلى منفذ معين.	إرسال طلبات TCP
udp	يرسل رسالة UDP إلى المضيف (host – ip address) الذي تم تكوينه و المنفذ .يدعم البث (broadcast) . مثل معظم العقد، يتم تهيئتها من خلال واجهة المستخدم أو خصائص الرسالة	
serial out	يرسل إلى المنفذ التسلسلي المحدد. يمكن تهيئتها لإرسال سطر جديد اختياري بعد أي حمولة رسالة .	

عقد Function :

تتضمن فئة عقد function مجموعة متنوعة من العقد التي تنفذ وظائف معالجة محددة. هذة المجموعة تمتد من عقد Delay و التبديل(Switch) إلى عقدfunction الخاصة بالبرمجة :

اسم العقدة	الوصف	أمثلة على العقد
function	عقدة function للبرمجة. يتم استخدام الجافا سكريبت (JavaScript) ، العقدة يمكن أن تكون مصممة لتنفيذ معالجة معقدة على الرسائل الواردة وتوليد رسالة او اكثر كمخرج.	بناء التدفق الأول (Hello World)تنبيهات الطقس إرسال طلبات TCP

template	عقدة القالب بشكل عام هي واجهة للمستخدم ، هذة العقدة تأخذ الرسالة المدخلة التي تحتوي على زوج من (الاسم:القيمة) و يتم درجهم في قالب. مفيدة لبناء الرسائل، HTML ، وتكوين الملفات .	
delay	العقدة بشكل عام هي التي تؤخر الرسائل إلى وقت محدد أو عشوائي. ويمكن أيضا أن يتم تهيئتها للتحكم في تدفق الرسائل (10 رسائل كل ثانية (.	
trigger	تنشئ رسالتين كمخرج مفصولتين بفترة زمنية معينة كلما تلقت رسالة مدخلة . كما يمكن استخدامها كمؤقت الحراسة (watchdog timer) .	استخدام التويتر للتحكم بالراسبيري باي
comment	عقدة التعليق يتم تهيئتها لتحتوي على عنوان و نص . وهي مفيدة جدا عند عمل تدفقات معقدة .	بناء التدفق الأول (Hello World)
http request	يسمح لك لإنشاء و إرسال طلب HTTP إلى عنوان URL معين . بطريقة (PUT , GET) تحتوي على رأس و حمولة payload ويتم تهيئتها من خلال واجهة المستخدم أو برمجيا.	
tcp request	هي عقدة طلب TCP بسيطة. ترسل msg.payload إلى خادم منفذ tcp وتتوقع الاستجابه. يمكن تهيئتها لانتظار البيانات، او حرف معين ، او العودة مباشرة.	إرسال طلبات TCP
switch	هذه العقدة تعمل على توجية الرسائل بناءا على خصائصهم. يتم تهيئة الخصائص باستخدام واجهة المستخدم ، حيث يمكنك استخدام المنطق (<،>،=< ....) على خصائص الرسالة.	استخدام عقدة التبديل لتوجية الرسالة
change	عقدة التغيير يمكن استخدامها لتعيين أو تغيير أو حذف خصائص الرسائل الواردة. وهناك مجموعة متنوعة من قواعد التهيئة التي تسمح بعمل تغيرات معقدة بما في ذلك البحث و الاستبدال في msg.payload .	التلاعب في محتوى الرسائل بإستخدام عقدة التغيير
range	عقدة Scaling من شأنها تعيين المدخلات الرقمية إلى مخرجات جديدة . وهي مفيدة لتحويل أو إحاطة نطاقات قيم المدخلات، مثل درجة الحرارة.	

اسم العقدة	الوصف	أمثلة على العقد
csv	هذه العقدة تقوم بتوزييع msg.payload و تحاول تحويل من/إلى CSV . إذا تلقت سلسلة، فإنها تخرج كائن جافا سكريبت ، واذا استقبلت كائن جافا سكريبت (JavaScript) ، فإنها تقوم بإخراج سلسلة CSV .	
html	استخراج العناصر من وثيقة HTML في msg.payload باستخدام محدد شكلي (CSS) . بالأساس يسمح لك لتحليل خروج HTML و إرجاع مجموعة من العناصر التي تتطابق.	
json	هذه العقدة تقوم بالتحويل من/إلى كائنJSON . إذا تمت استقبال كائن جافا سكريبت (JavaScript) ، فتقوم بإخراج النواتج JSON . أما إذا تلقت JSON ، فتقوم بإخراج كائن جافا سكريبت (JavaScript) .	تلقي JSON عبر رسالة MQTT
xml	هذه العقدة تقوم بالتحويل من/إلى صيغة XML . إذا تلقت كائن جافا سكريبت (JavaScript) ، فتقوم بإخراج سلسلة XML، أما إذا استقبلت سلسلة XML ، فتقوم بإخراج كائن جافا سكريبت (JavaScript) .	
rbe	تقوم بتوليد الرسالة فقط إذا طرأ تغيير مختلف على المدخلات السابقة (سلسلة أو رقم)، أو إذا تغيرت المدخلات بمقدار معين (في حال وضع deadband) – فقط للأرقام .	استخدام عقدة RBE

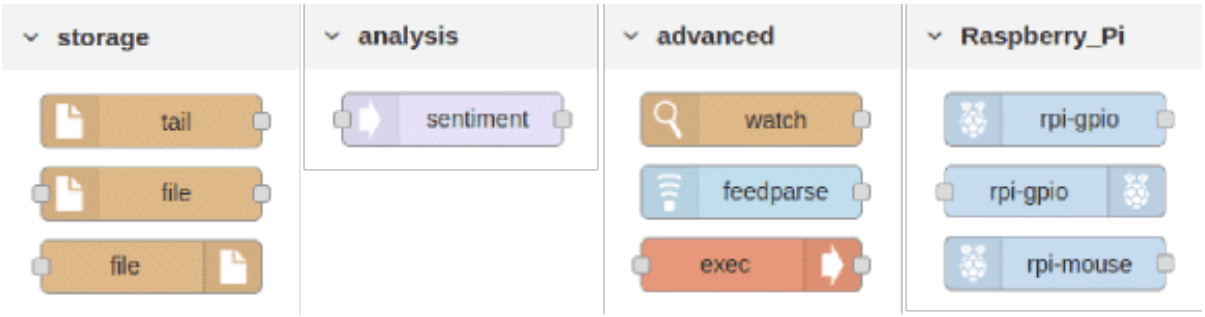
عقد Social :

عقد وسائل الاعلام الاجتماعية الأساسية تدعم التفاعل مع البريد الإلكتروني و التويتر. فإنها تتيح للتدفقات لإرسال أو استقبال رسائل البريد الإلكتروني ، أو إرسال أو تلقي تويت.

اسم العقدة	الوصف	أمثلة على العقد
email in	يمكن تهيئتها لقراءة عدة مرات من خادم IMAP وإرجاع بريد إلكتروني جديد . وضع msg.topic إلى موضوع البريد الإلكتروني و إما وضع msg.payload إلى محتوى نص البريد الإلكتروني أو msg.html إذا كان البريد الإلكتروني هو HTML .	
twitter in	يقوم بإرجاع تغريدة (tweets) كرسائل . يمكن استخدامها للبحث في العام أو على حساب المستخدم عن تغريدات تحتوي على مصطلح البحث الذي تم تهيئته بواسطة المستخدم أو رسائل المباشرة الواردة من قبل المستخدم.	استخدام التويتر للتحكم بالراسبيري باي
email out	يرسل رسالة واردة كرسالة بريد إلكترونيعبر خادم IMAP الذي تم إعداده. وسيتم تحويل البيانات الثنائية (binary data) إلى المرفق.	تنبيهات البريد الالكتروني
twitter out	يقوم بتغريد msg.payload على الحساب الذي يتم إعداده. يمكن ارسال الرسائل المباشرة و سيتم ارسال البيانات الثنائية كصورة .	تنبيهات الطقس

عقد التخزين :

العقدة الإفتراضية المحددة للتخزين، لأنها تستهدف أجهزة مثل الراسبيري باي المحدودة للغاية ، و تركز على تخزين الملف .



ملاحظة : FRED لأنه عبارة عن خدمة سحابية ، فإنه لا يدعم عقد الملفات الأساسية. وبدلا من ذلك يستبدل هذه مع مجموعة متنوعة من عقد التخزين تتراوح بين Mongo إلى Dropbox . ومع ذلك ، للتاكد و تغطية هذة العقد الإفتراضية قم بثبيت Node-RED على جهازك .

اسم العقدة	الوصف
tail	تضيف ذيل إلى الملف الذي تم إعداده. تعمل على انظمة (Linux/Mac) فقط . هذه العقدة لا تعمل على ملف انظمة ويندوز، كما أنه يعتمد على الامر tail -F.
file in	يقرأ الملف المحدد و يرسل المحتوى كـ msg.payload ، واسم الملف كـ msg.filename . يمكن تهيئة اسم الملف في العقدة. إذا ترك فارغا، فإنه يجب تعيين على msg.filename في الرسالة الواردة .
file	تقوم بكتابة msg.payload إلى الملف المحدد . يمكن تهيئة اسم الملف في العقدة. إذا ترك فرغا، فإنه يجب تعيينه على msg.filename في الرسالة الواردة . السلوك الافتراضي لها هي الإلحاق إلى الملف. ويمكن تغيير ذلك إلى الكتابة فوق الملف كل مرة . على سبيل المثال ، إذا كنت ترغب في إخراج صفحة ويب ثابته أو تقرير .

عقد التحليل (Analysis nodes):

عقد التحليل تنفذ التحاليل القياسية على الرسائل الواردة. العقدة الوحيدة الت قدمت هي عقدة sentiment ،والتي يمكن استخدامها في المحاولة و لتحديد الـ sentiment للرسالة الواردة على أساس الكلمات المستخدمة في الرسالة .على سبيل المثال ، البريد الإلكتروني أو التويتر .

اسم العقدة	الوصف	أمثلة على العقد
sentiment	عقدة sentiment تقوم بتحليل msg.payload و تسجل نتيجة مشاعر الرسالة على أساس تحليل الكلمة. وتضيف كائن msg.sentiment الذي يحتوي على الناتج AFINN-111 كـ msg.sentiment.score . و النتيجة عادة تتراوح من 5– إلى 5 .	

العقد المتقدمة (Advanced nodes) :

وهناك مجموعة من العقد المتنوعة التي تقدم أنواع مختلفة من الوظائف.

اسم العقدة	الوصف
watch	تراقب دليل أو ملف لإجراء التغييرات. يمكن إدخال قائمة من الدلائل مفصولة بفواصل أو الملفات. وسوف تحتاج إلى وضع علامات اقتباس "....." حول أي مسافات. على نظام الويندوز، يجب عليك استخدام خطوط مائلة مزدوجة // في جميع اسماء الدليل. يتم وضع إسم الملف كاملا للملف الذي تغير فعلا في msg.payload ، في حين يتم استرجاع نسخة stringfied من قائمة المراقبة في msg.topic . يحتوي الـ msg.file على اسم قصير للملف الذي تغير. msg.type يحتوي على نوع التغير، عادة اما ملف او دليل، في حين msg.size يحمل حجم الملف بالبايت (bytes) .
feedparse	هذه العقدة تراقب RSS/atom feed للمدخلات الجديدة و توفر إدخالات جديدة كالرسائل . ويمكن تهيئتها للاستعلام عن التغذية في فترة زمنية محددة .
exec	يستدعي أمر النظام ويوفر ثلاث مخرجات : stdout, stderr, return code . افتراضيا يستخدم exec()، الذي يستدعي الأمر ، وينتظر الانتهاء ، ومن ثم إرجاع النتيجة كاملة دفعة واحدة ، جنبا إلى جنب مع أي أخطاء.

عقد الراسبيري باي (Raspberry Pi nodes) :

اسم العقدة	الوصف	أمثلة على العقد
rpi_gpio in	عقدة الراسبيري باي الخاصة بالادخال. تولد msg.payload إما 0 أو 1 ، وهذا يتوقف على حالة إدخال الدبوس (pin) . يمكنك ايضا تفعيل المدخل pullup resistor أو pulldown resistor. تم تعيين msg.topic إلى pi/{رقم Pin} يتطلب مكتبة RPi.GPIO بايثون نسخة 0.5.8 (أو الأحدث) من أجل العمل . ملاحظة : نحن نستخدم أرقام الـ pin الفعلية على موصل P1 كما انها أسهل لتحديد الموقع.	

rpi_gpio out	عقدة الراسبيري باي الخاصة بالإخراج. تتوقع msg.payload مع إما 0 أو 1 (true أو false) . وسوف يتم تعيين الدبوس إما HIGH أو LOW اعتمادا على القيمة التي يتم تمريرها . ويمكن أيضا تعيين القيمة الأولية للدبوس وقت نشر التدفق إما 0 أو 1 . عند إستخدام PWM ، فهي تتوقع قيمة المدخلات بين 0–100 . يتطلب مكتبة RPi.GPIO بايثون نسخة 0.5.8 (أو الأحدث) من أجل العمل .	
rpi_mouse	عقدة راسبيري باي زر الماوس. يولد msg.payload مع إما 1 أو 0 عندما يتم الضغط على زر الماوس المحدد ثم تحريره. يحدد أيضا msg.button إلى قيمة الرمز ، 1=اليسار، 2=اليمين ، 4= الوسط حتى تتمكن من العمل خارج أي زر او مجموعة تم الضغط عليها.	