n8n

n8n بالعربي

دليلك العملي لتعلّم الأتمتة من الصفر

مينا سعد

n8n

جدول المحتويات

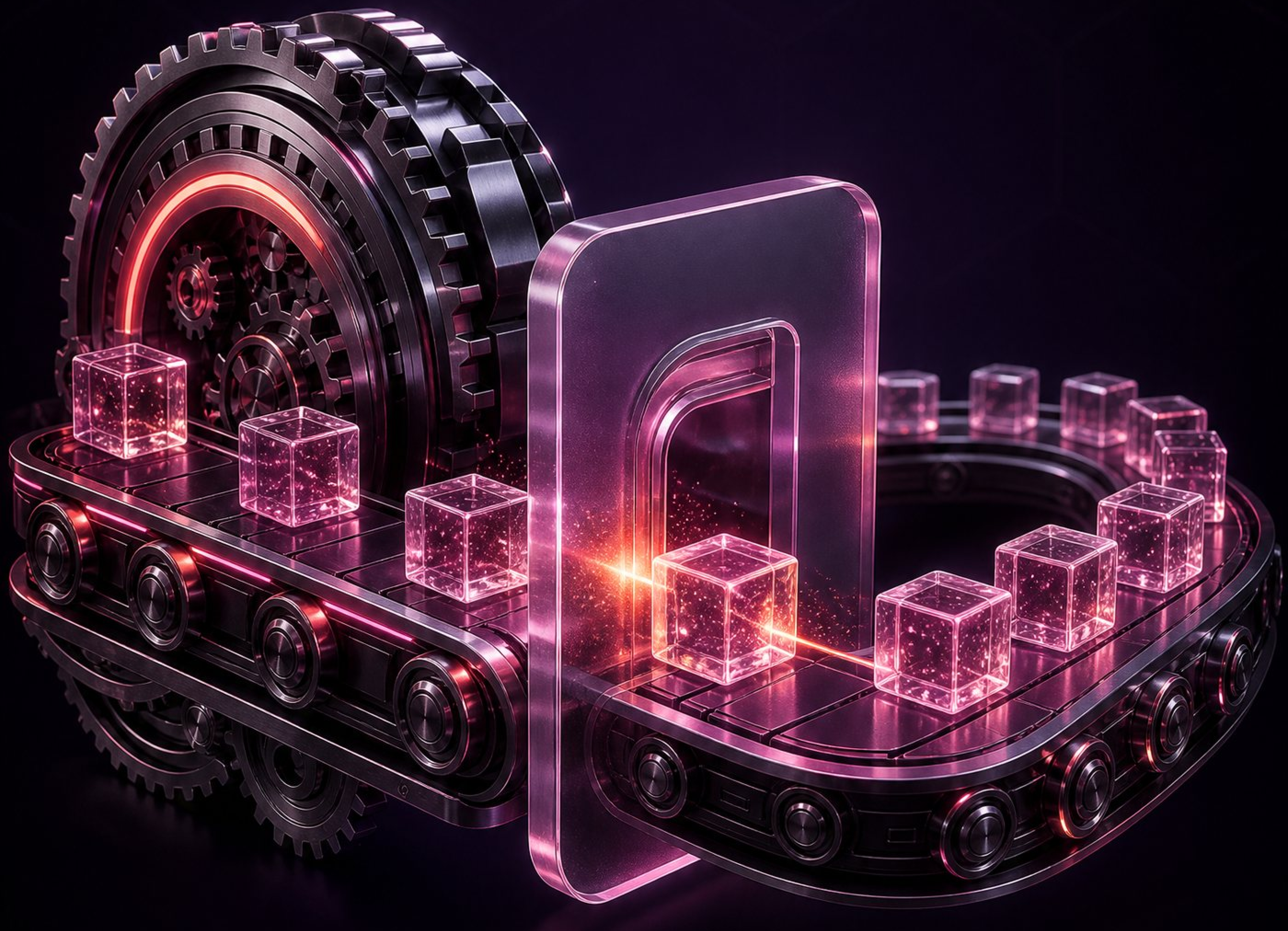
1. ما هي الأتمتة ولماذا n8n؟
2. كيف تبدأ: السحابة مقابل الاستضافة الذاتية
3. جولة في واجهة n8n
4. المفاهيم الأساسية: Workflow و Node و Connection
5. أنواع الـ Nodes: المُحفِّزات والإجراءات والعقد الأساسية
6. المُحفِّزات: يدوي وجدولة و Webhook
7. البيانات في n8n: العناصر (Items) و JSON
8. التعبيرات: قراءة البيانات وتحويلها
9. عقد المنطق: IF و Switch و Filter
10. تشكيل البيانات: Edit Fields و Merge والتكرار
11. الاتصال بالخدمات: HTTP Request و App Nodes
12. بيانات الاعتماد والأمان الأساسي
13. معالجة الأخطاء الأساسية
14. أفضل الممارسات وتنظيم الـ Workflows
15. ثلاثة أمثلة عملية كاملة
16. الختام بالذكاء الاصطناعي والخطوات التالية

الفصل 1

ما هي الأتمتة ولماذا n8n؟

اجعل الآلة تكرّر العمل بدلا منك

الأتمتة: الآلة تكرّر بدلا منك



خطوة تتكرر كل يوم؟ دعها للآلة

تستقبل حدثًا، تعالجه، ثم تنفّذ الإجراء

بلا تعب، وبلا نسيان، وبلا خطأ بشري

ما هي n8n؟

منصة بصرية لأتمتة سير العمل

n8n

الفصل 2

كيف تبدأ: السحابة أم الاستضافة الذاتية

طريقتان لتشغيل n8n

اختر مسارك، ثم افتح المحرر

طريقتان لتشغيل n8n

السحابة (n8n Cloud):

بدون تثبيت ولا صيانة

الاستضافة الذاتية

(self-hosted):

عبر npm أو Docker

Docker | npm

السحابة أسهل، والاستضافة الذاتية أكثر تحكما

أيهما تختار؟ موازنة سريعة

اختر السحابة للبدء فورا دون إدارة خادم

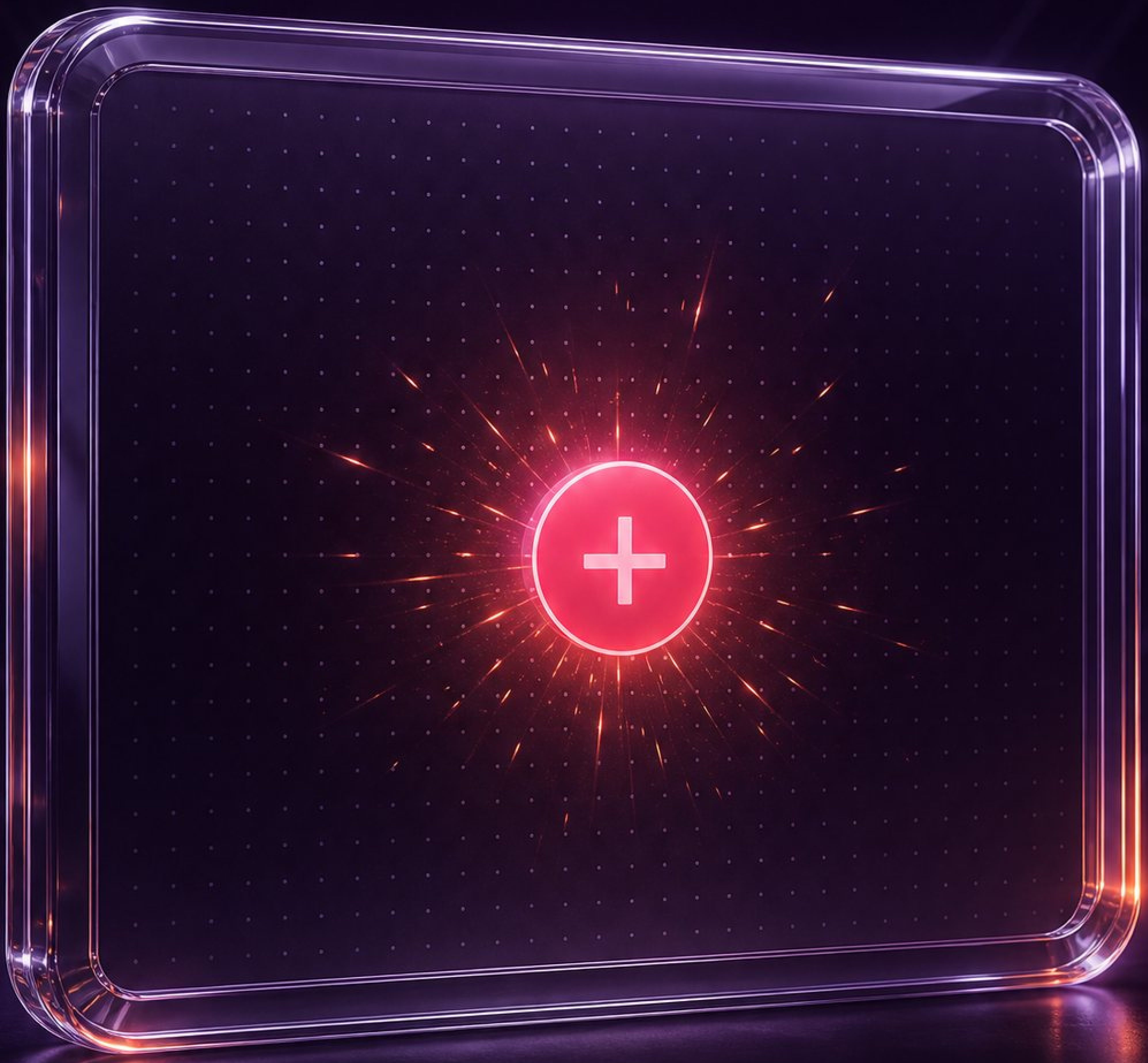
السحابة: تجربة 14 يوما بميزات خطة Pro

اختر الاستضافة الذاتية لتحكم كامل في البيانات

الذاتية: مجانية كنسخة مجتمع (Community)

افتح المحرر وأنشئ سير عملك الأول

New Workflow



السحابة: ادخل على yourname.app.n8n.cloud

أنشئ سير عمل جديدا من زر الإضافة

تبدأ بلوحة فارغة جاهزة للبناء

ماذا تعلّمت في هذا الفصل

السحابة أسهل، والاستضافة الذاتية أكثر تحكما

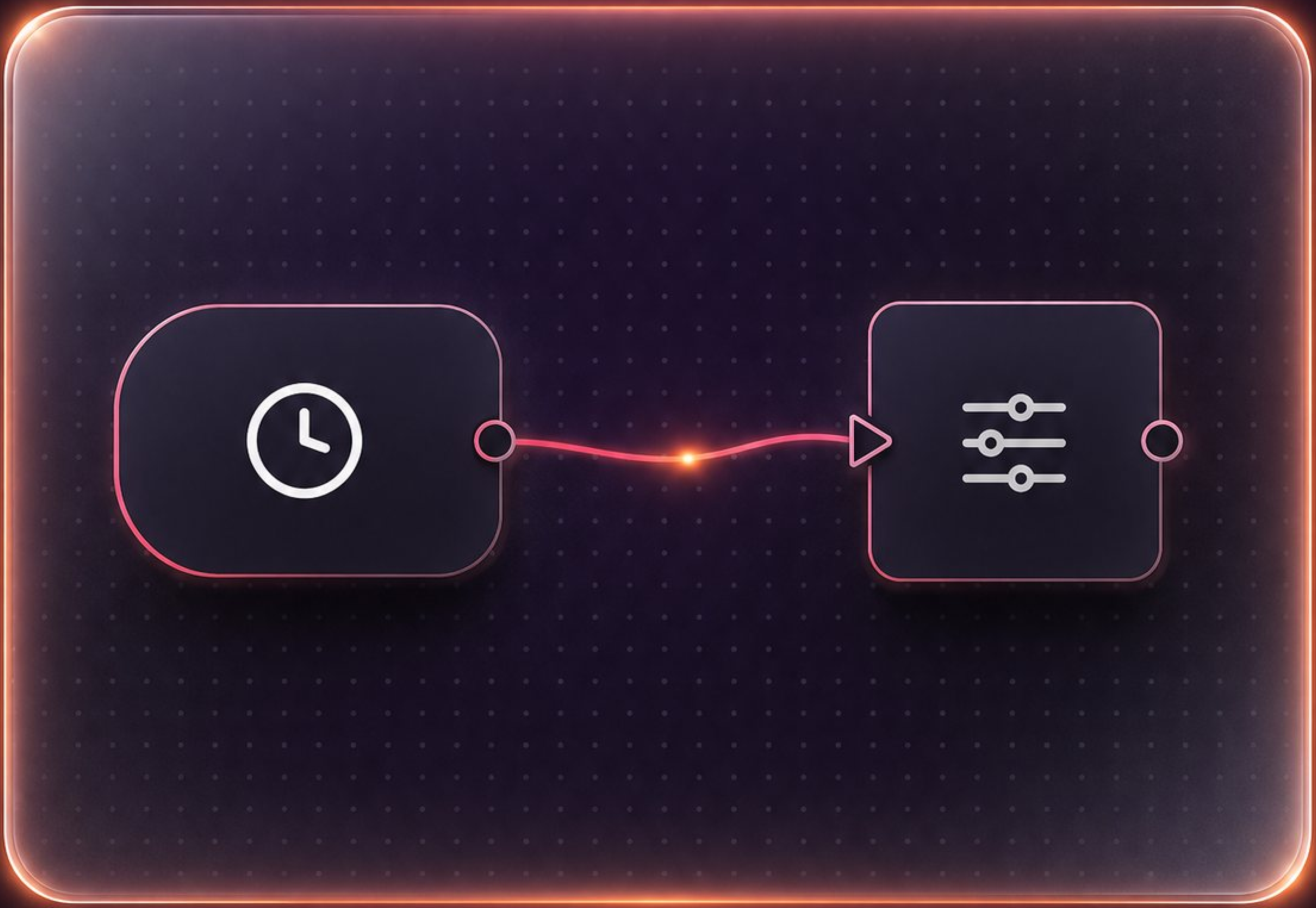
هناك مساران: السحابة والاستضافة الذاتية

كل شيء يبدأ من سير عمل فارغ

الوصول للمحرر متطابق في الحالتين

n8n

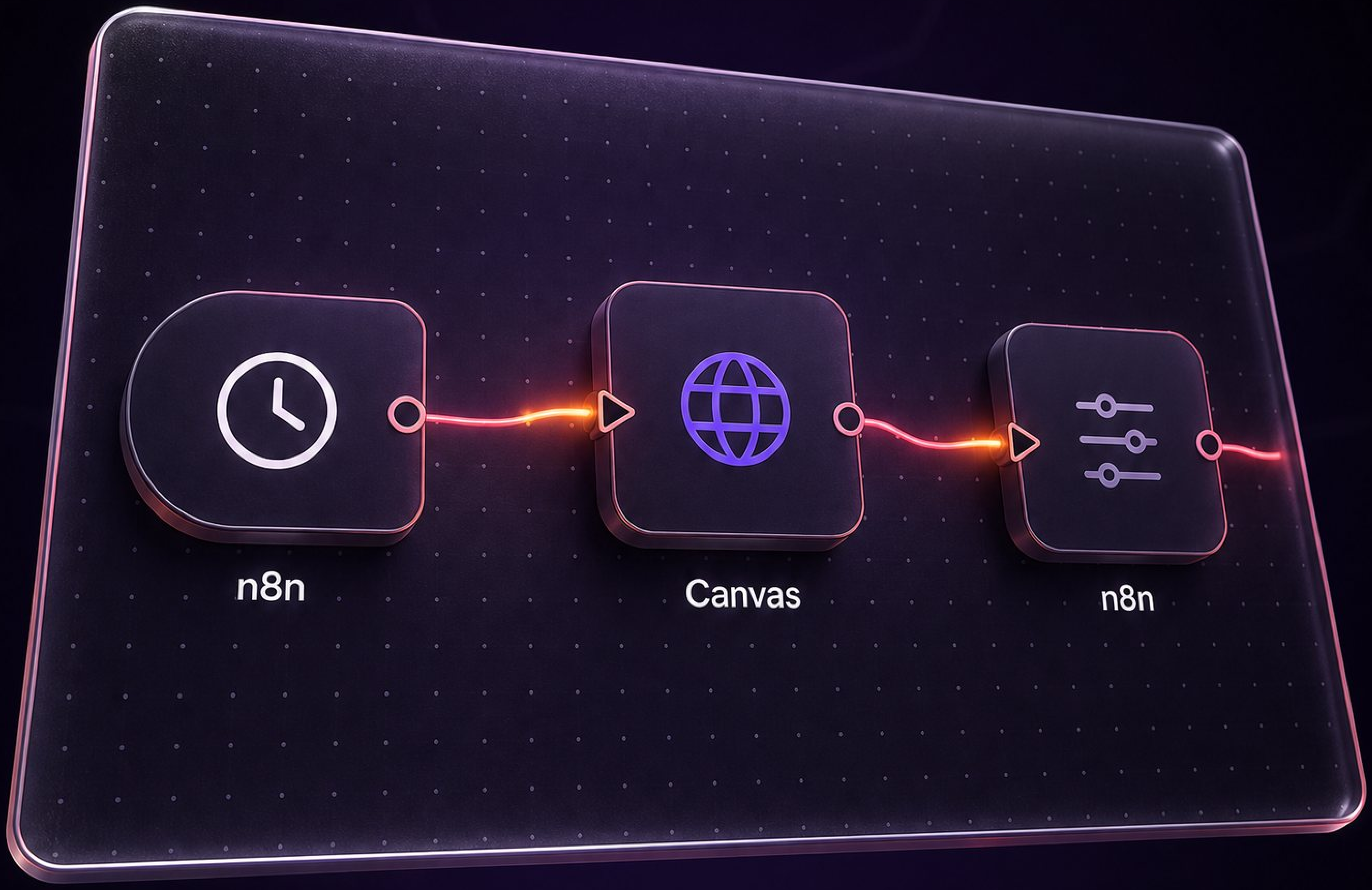
الفصل 3



جولة في واجهة n8n

تعرّف على اللوحة وقائمة العقد والتشغيل

اللوحة: مساحة بناء سير العمل



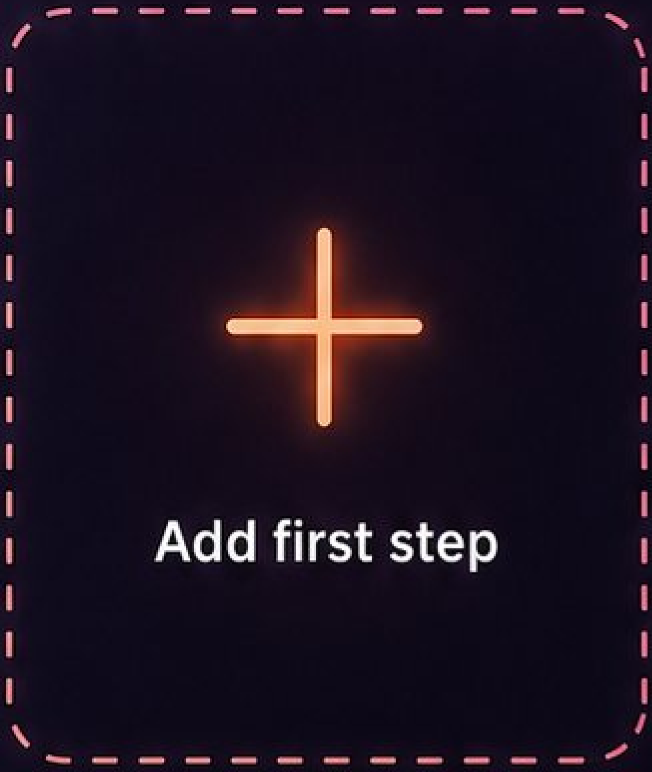
اللوحة (Canvas) هي مساحة العمل المرئية

العقد بطاقات، والوصلات خيوط تربطها

تسحب وتكبّر وترتّب بحرية

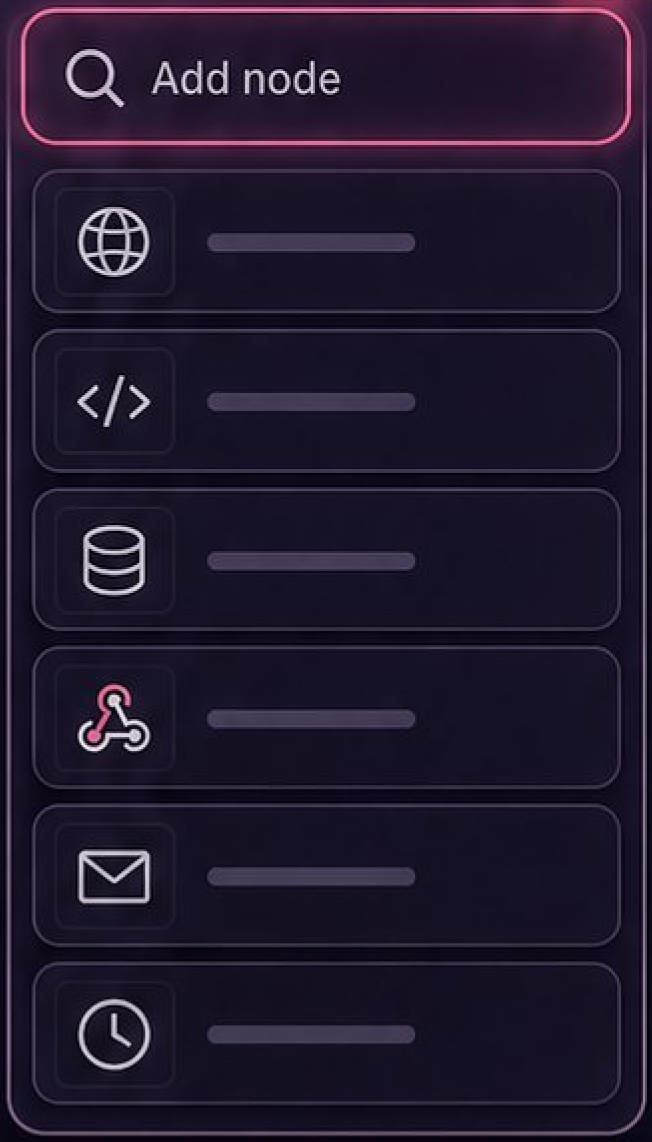
إضافة العقد وربطهـا

1



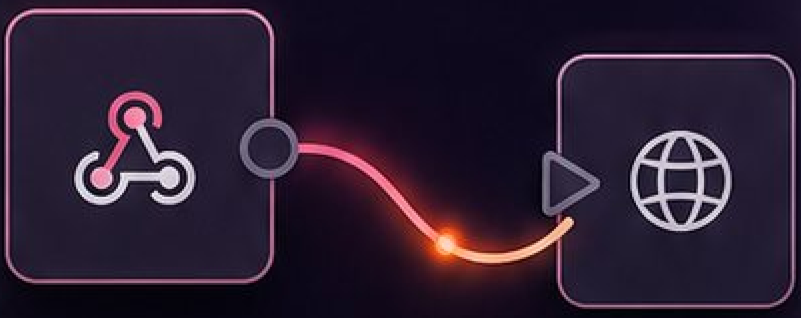
لأول عقدة اضغط
«Add first step» jj

2



ابحث عن العقدة
أو تصفَّحها ثم اخترها

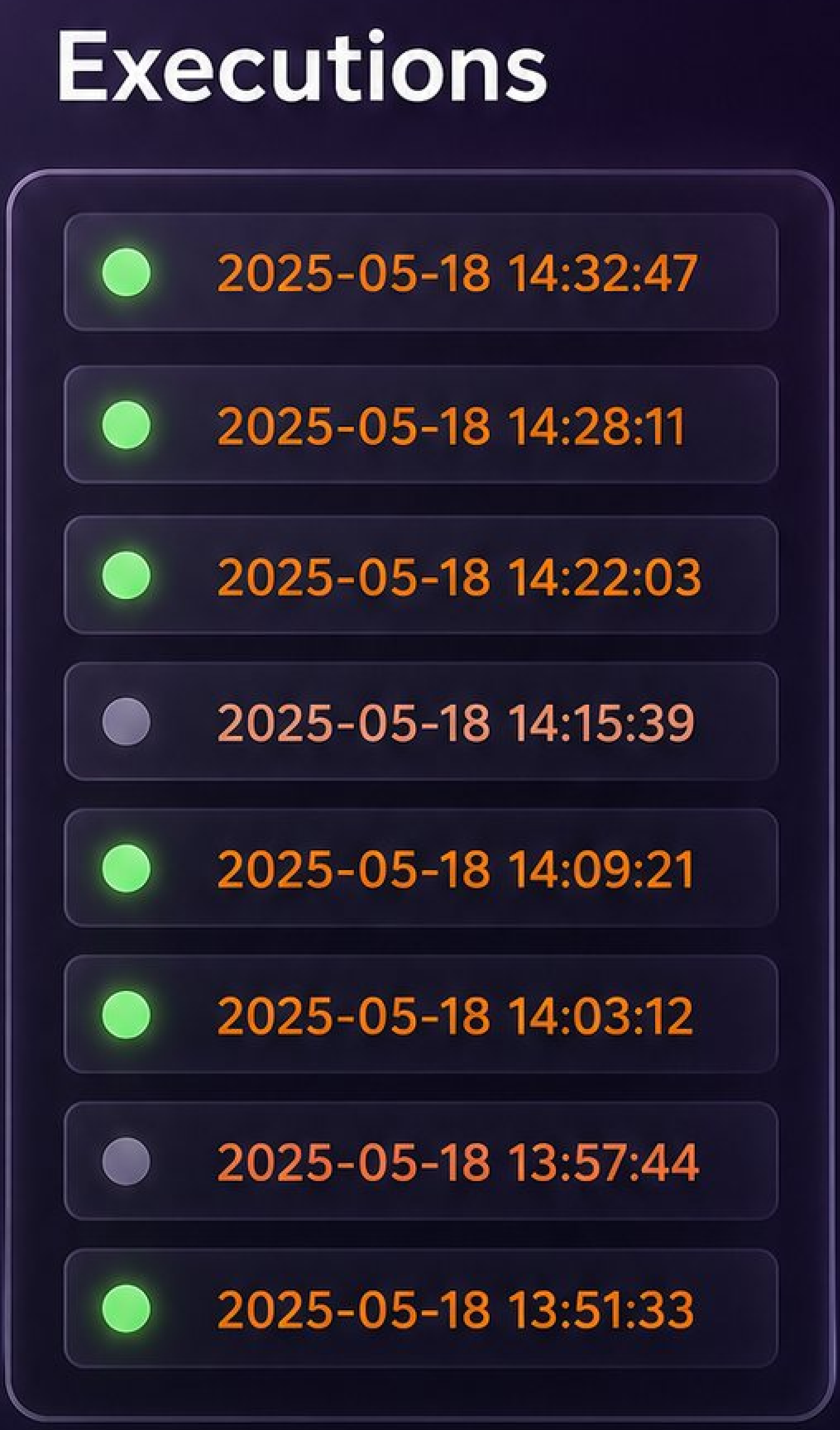
3



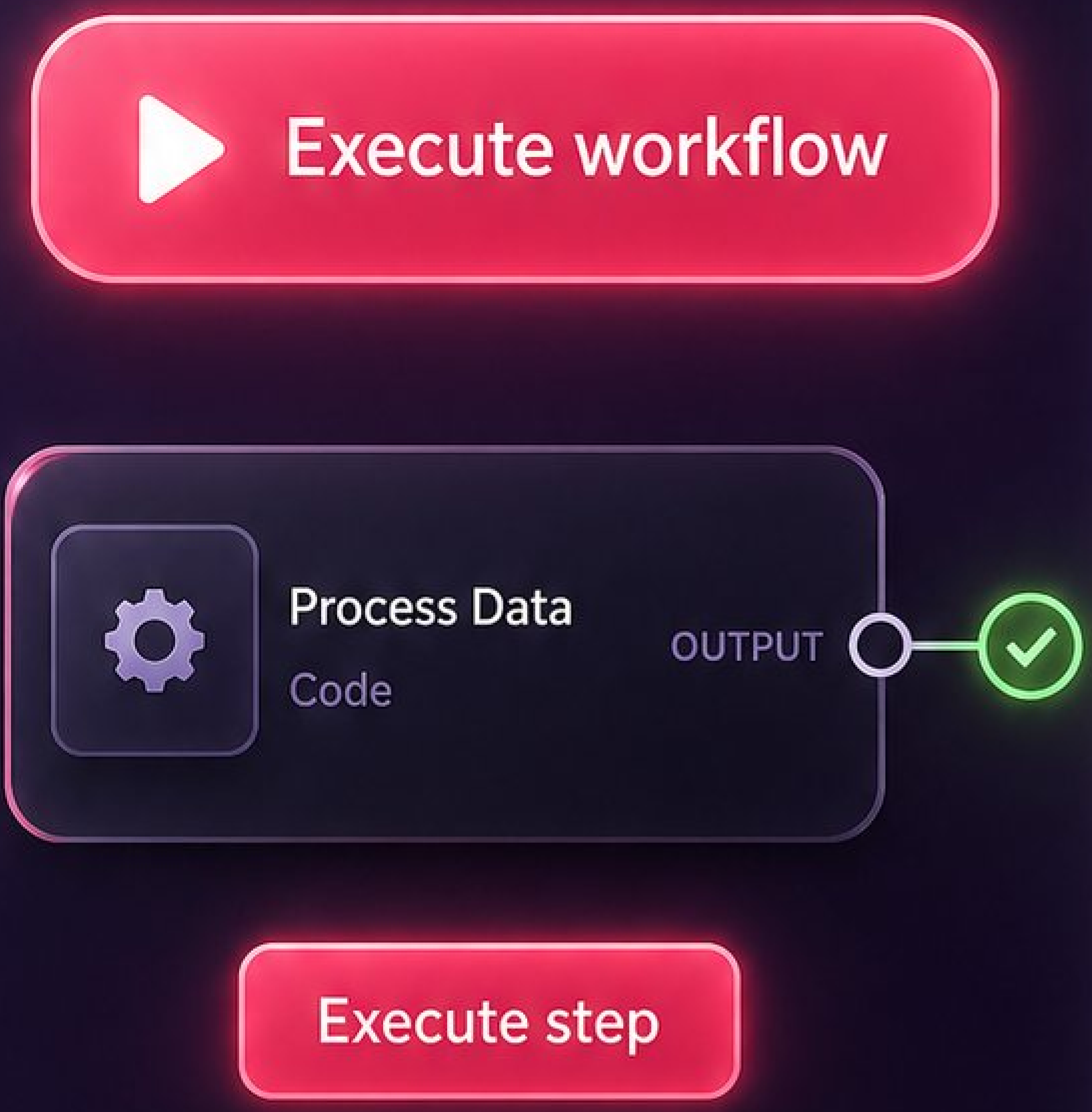
اسحب من النقطة الرمادية
إلى مدخل العقدة التالية

لإضافة عقدة تالية
اضغط موصّل «Add node»

التشغيل ومتابعة السجل



سجل «Executions»
يحفظ تاريخ التشغيلات

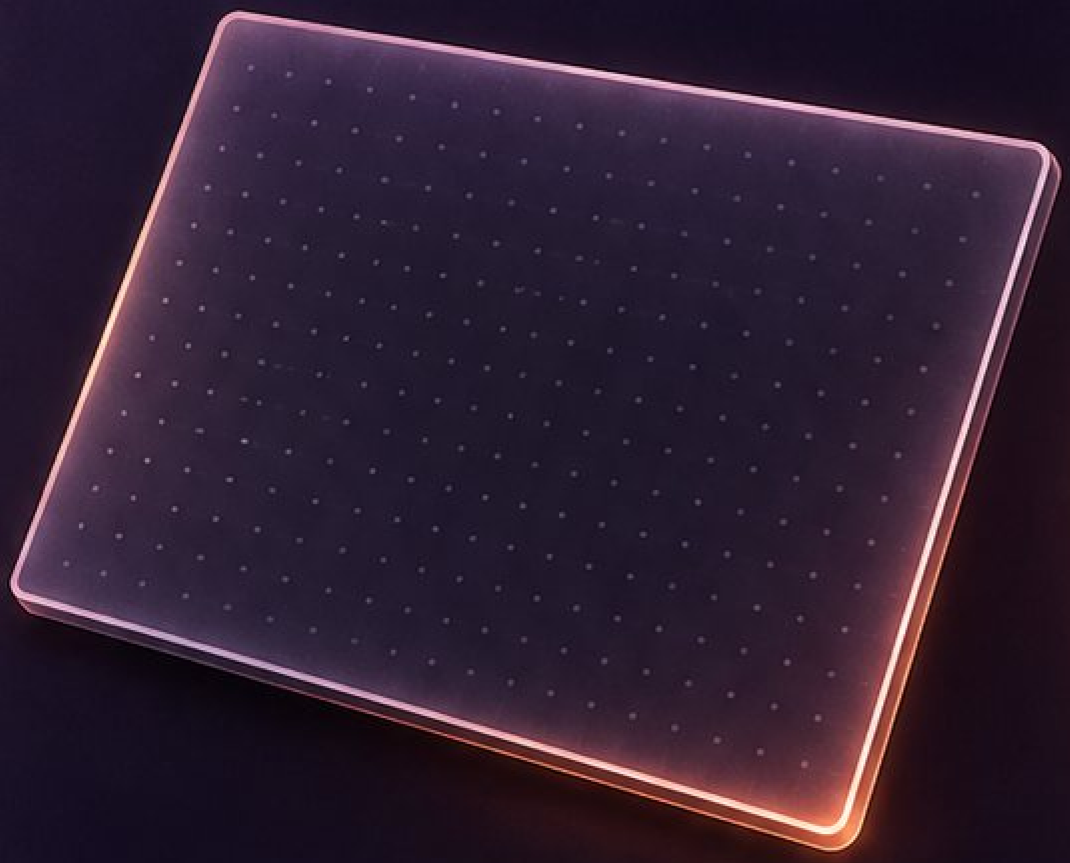


زر «Execute workflow»
يشغل سير العمل للاختبار

«Execute step»
يشغل عقدة واحدة فقط

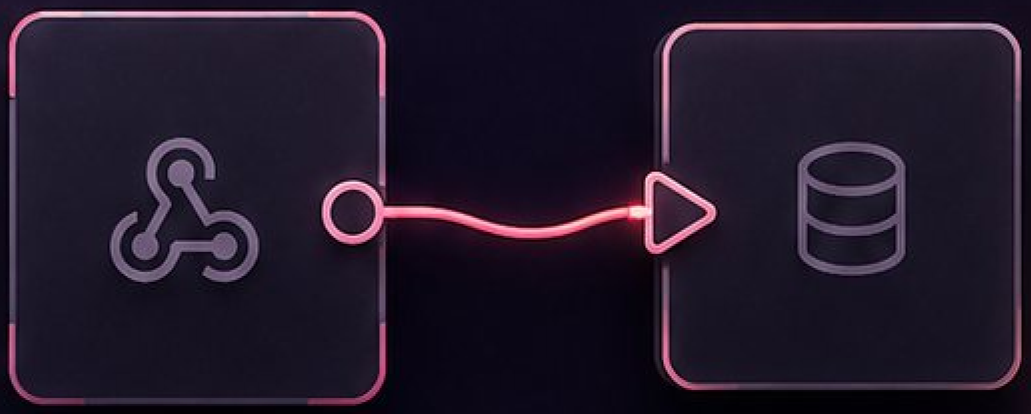
كل تشغيل يظهر مخرجاته
داخل العقد

ماذا تعلّمت في هذا الفصل



اللوحة (Canvas) مساحة بناء
سير العمل بصريا

قائمة العقد تُفتح بزر
«Add node» أو «Add first step»



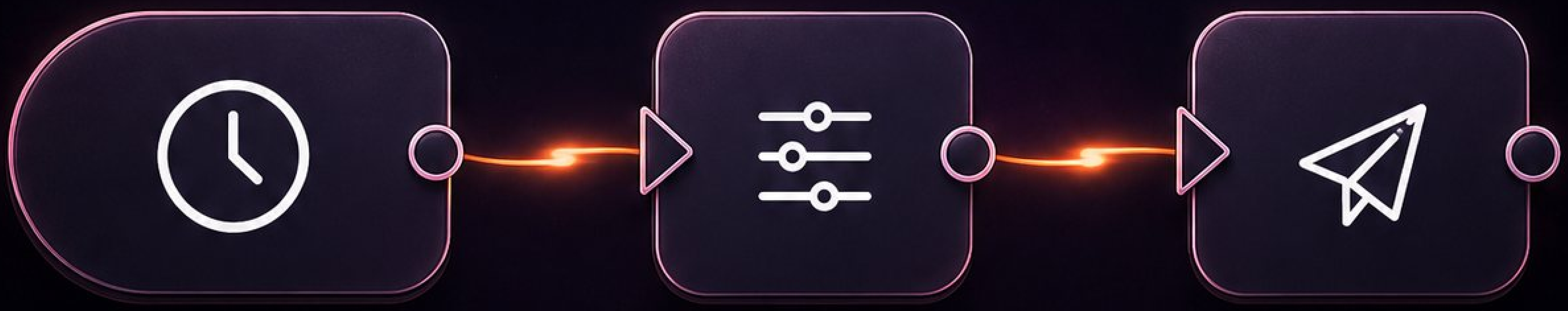
الوصلة تربط مخرج عقدة
بمدخل التالية

يختبر, «Execute workflow»
والسجل يحفظ التشغيلات

n8n

الفصل 4

سير العمل والعقدة والوصلة

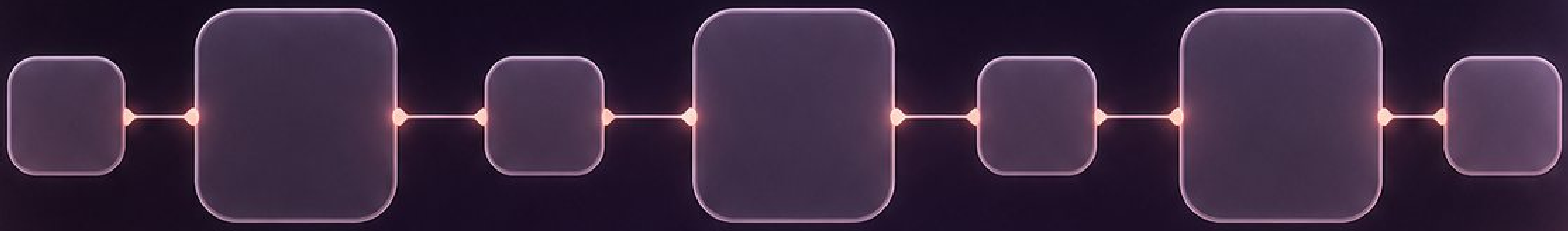


ثلاثة مفاهيم تبني عليها كل شيء

عقد متصلة، وبيانات تتدفق بينها

ثلاثة مفاهيم في صورة واحدة

سير العمل (Workflow): اللوحة الكاملة



العقدة .(Node):
خطوة واحدة محددة

الوصلة (Connection):
خيط يمرر البيانات

عقد متصلة = سير عمل واحد

العقدة تؤدي مهمة واحدة

العقدة (Node): مدخل أو معالجة أو مخرج

مدخل على اليسار، مخرج على اليمين

كل عقدة تعالج كل عنصر (Item) على حدة

سلسلة من ثلاث عقد تتدفق يسارا فيمينا

1) مُحفّز (Trigger) يبدأ التشغيل

2) عقدة تعالج البيانات

3) عقدة ترسل النتيجة

البيانات تتدفق من المخرج إلى المدخل التالي

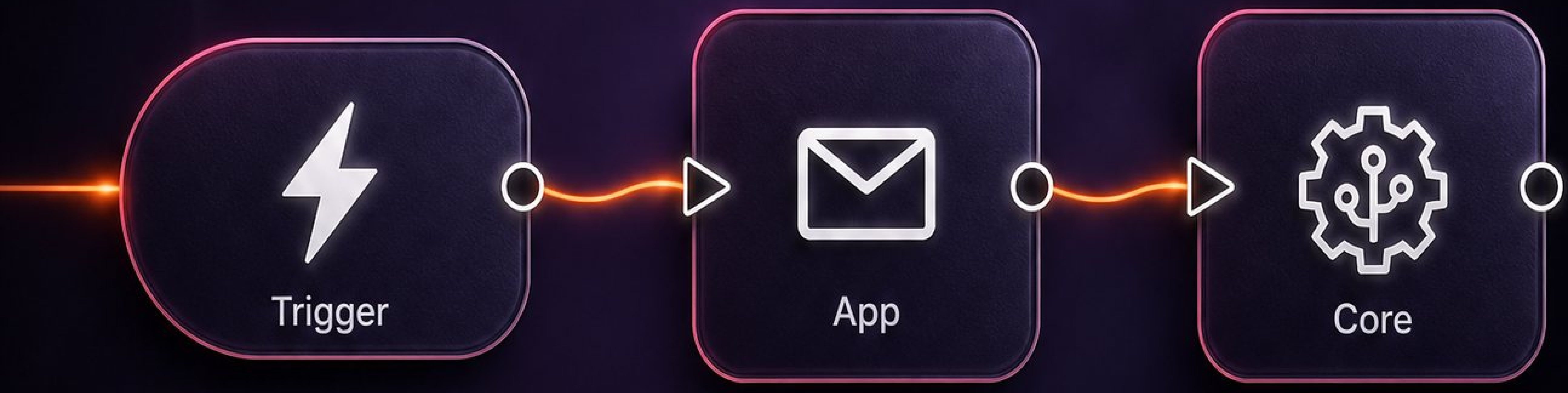
ماذا تعلّمت: عقد + وصلات = سير عمل

الفصل 5

أنواع العُقّد: المُحفّزات والإجراءات والعقد الأساسية

كل عقدة لها وظيفة، ولكل نوع شكله على اللوحة.

ثلاث عائلات من العُقَد



العقدة الأساسية (Core):
منطق وجدولةً وطلبات عامةً

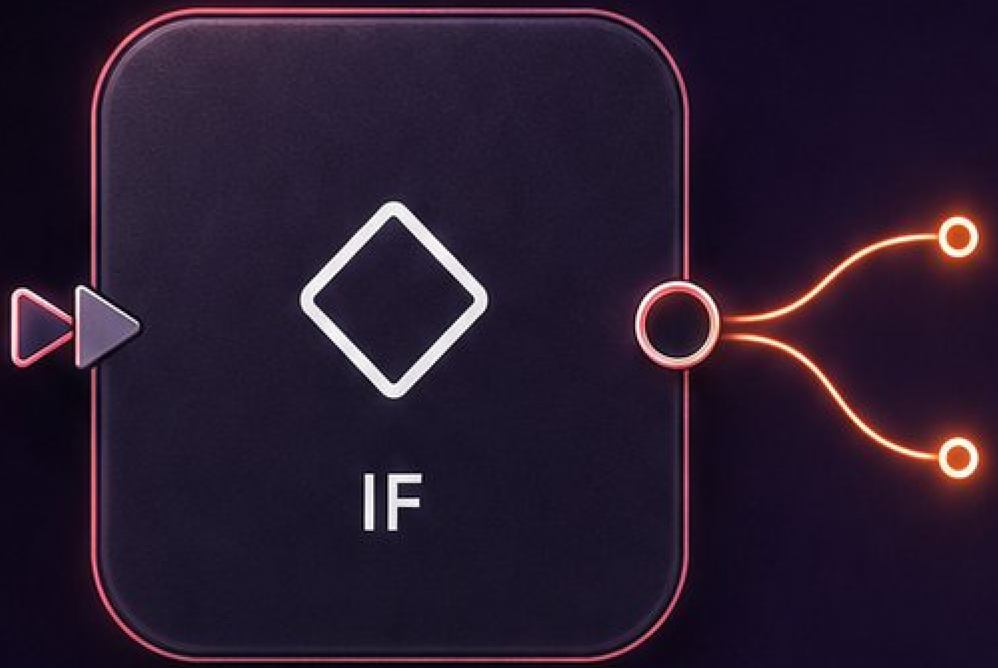
عقدة التطبيق (App):
تتحدث إلى خدمة مثل Gmail

المُحفِّز (Trigger):
يبدأ التشغيل

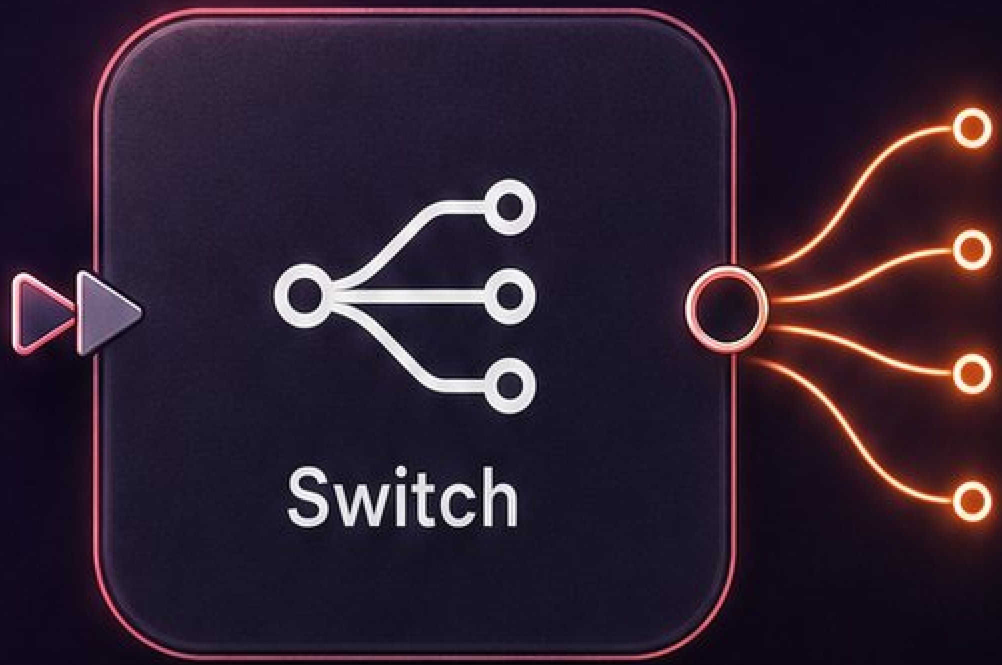
كل سير عمل يبدأ بمُحفِّز واحد على الأقل

العُقَد الأساسية: صندوق أدواتك

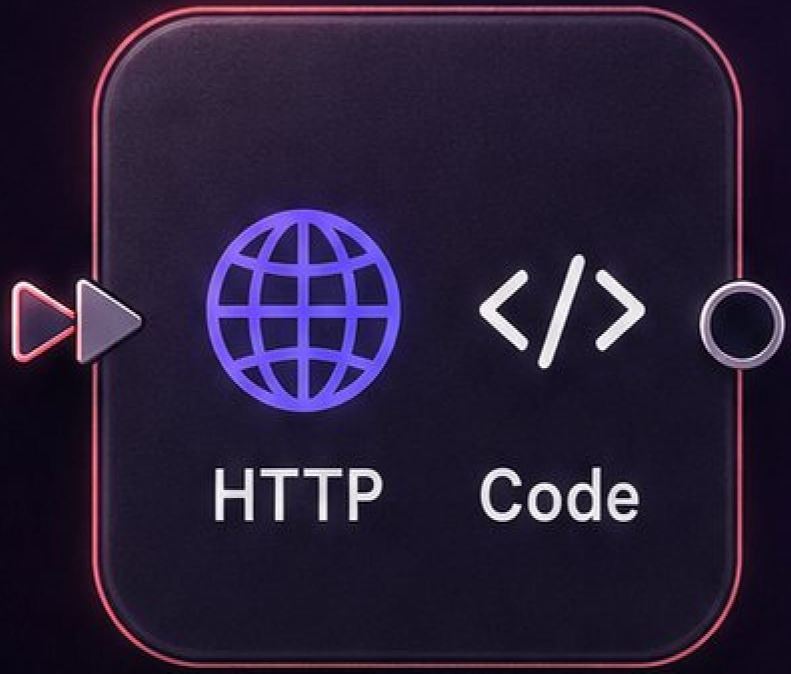
تعديل الحقول (Edit Fields):
يضيف ويعدّل البيانات



الشرط (IF):
مساران، صحيح وخطأ

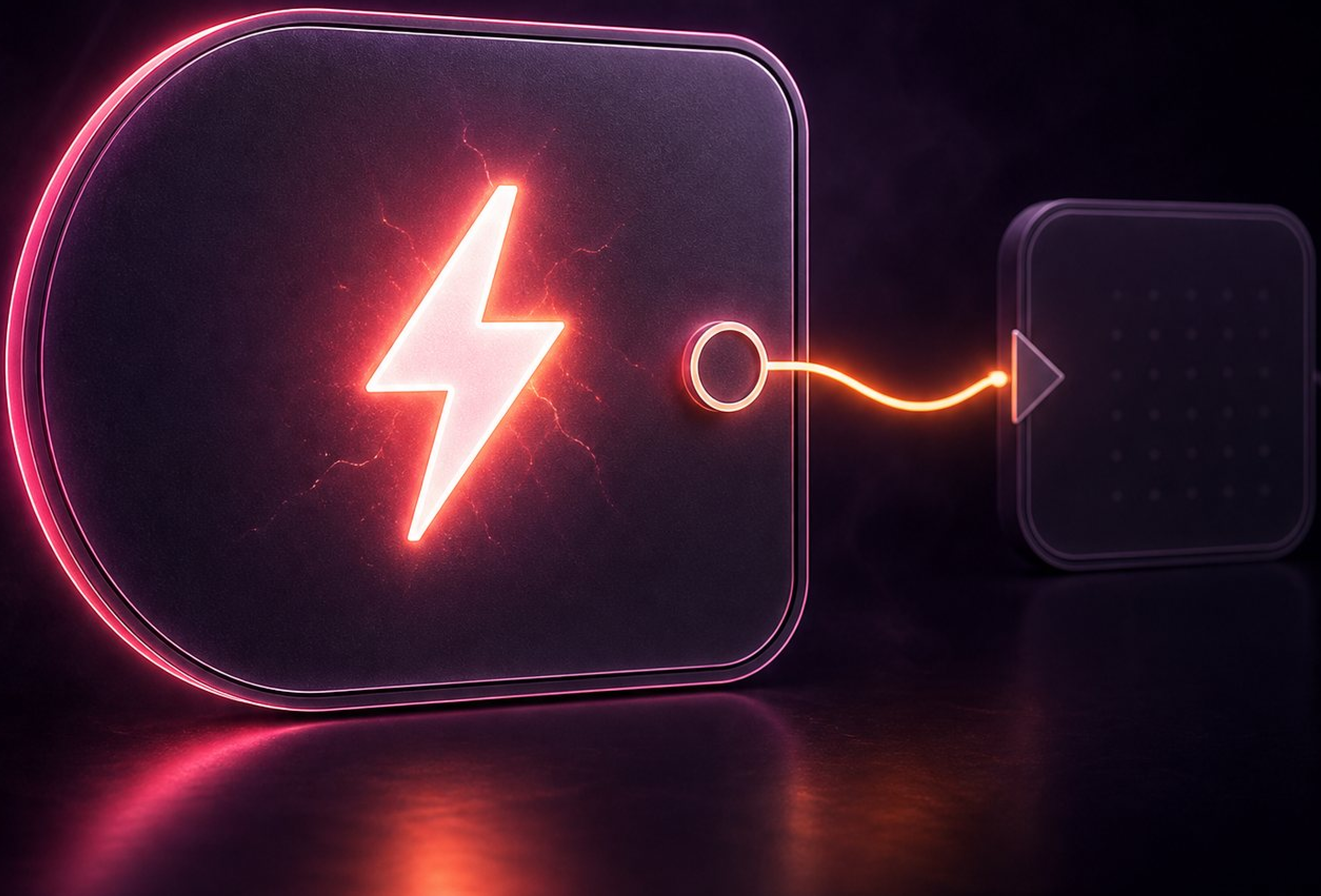


التوجيه (Switch):
مسارات إخراج متعددة



طلب HTTP والكود (Code):
مرونة لأي حالة

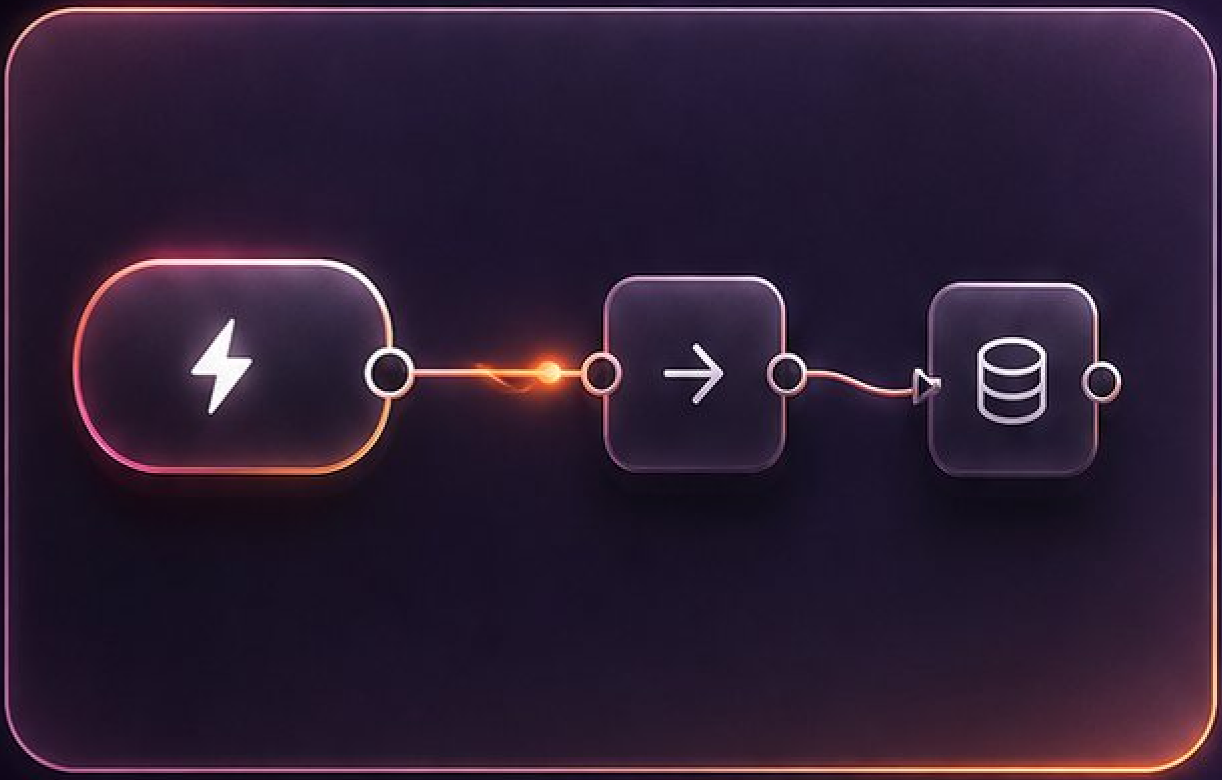
كيف تميّزها على اللوحة؟



- المُحفِّز: أيقونة برق ويبدأ من اليسار
- المُحفِّز لا يستقبل وصلة قبله
- ابحث باسم الخدمة، ثم اختر إجراء أو مُحفِّز

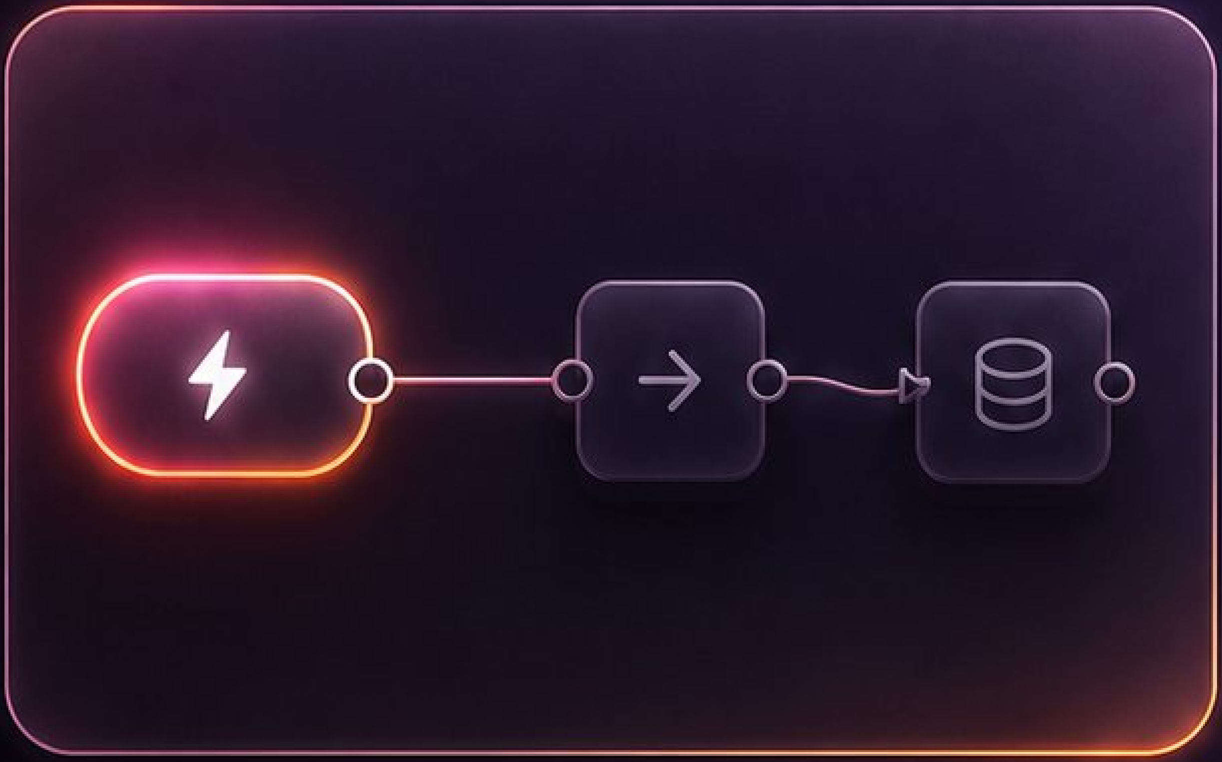
ماذا تعلّمت في هذا الفصل

ثلاث عائلات: مُحفّز، تطبيق/إجراء، أساسية



المُحفّز يبدأ، والباقي يَنَفّذ ويعالج

العقد الأساسية لا ترتبطِ بخدمة واحدة



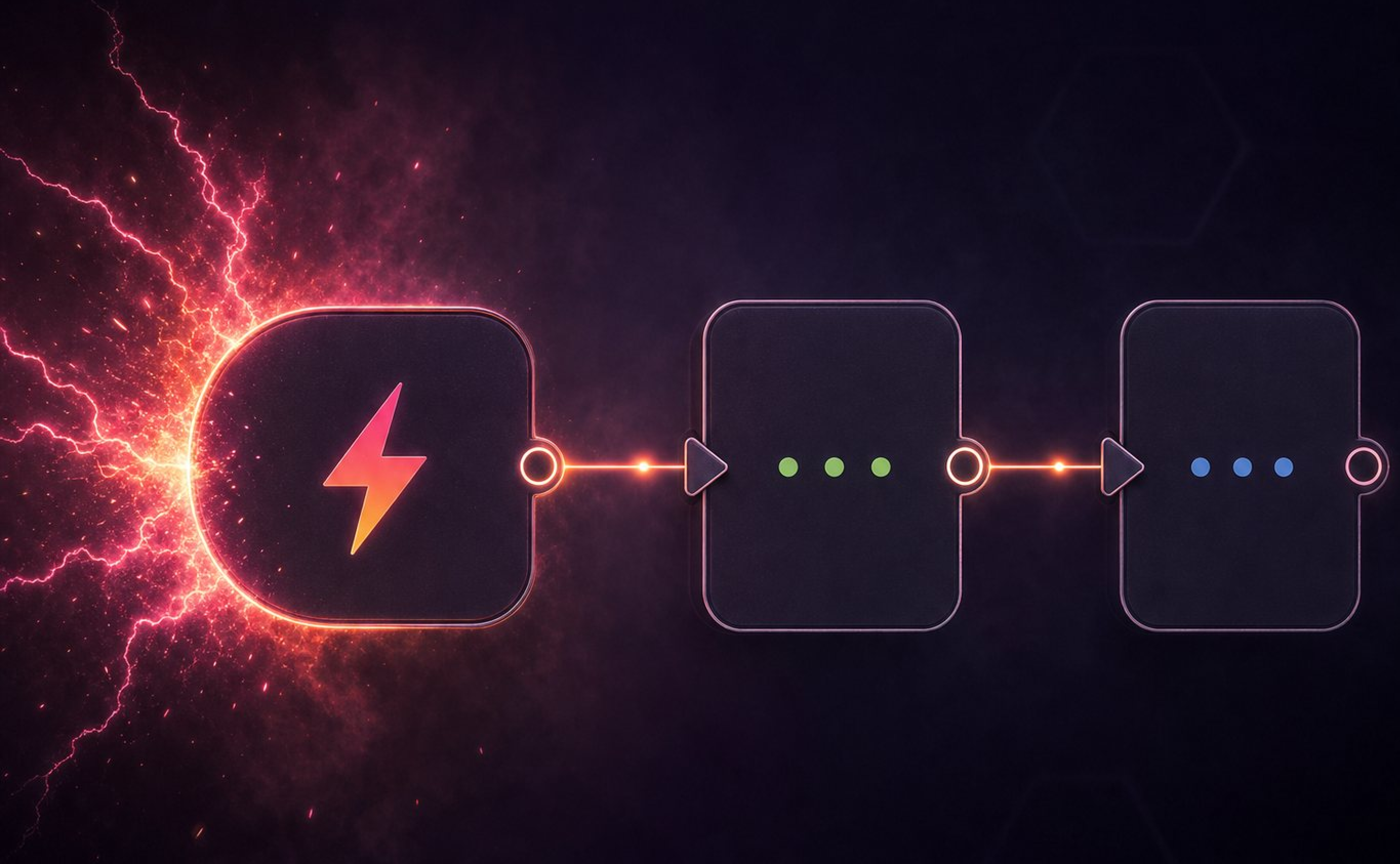
أيقونة البرق والموضع يكشفان المُحفّز

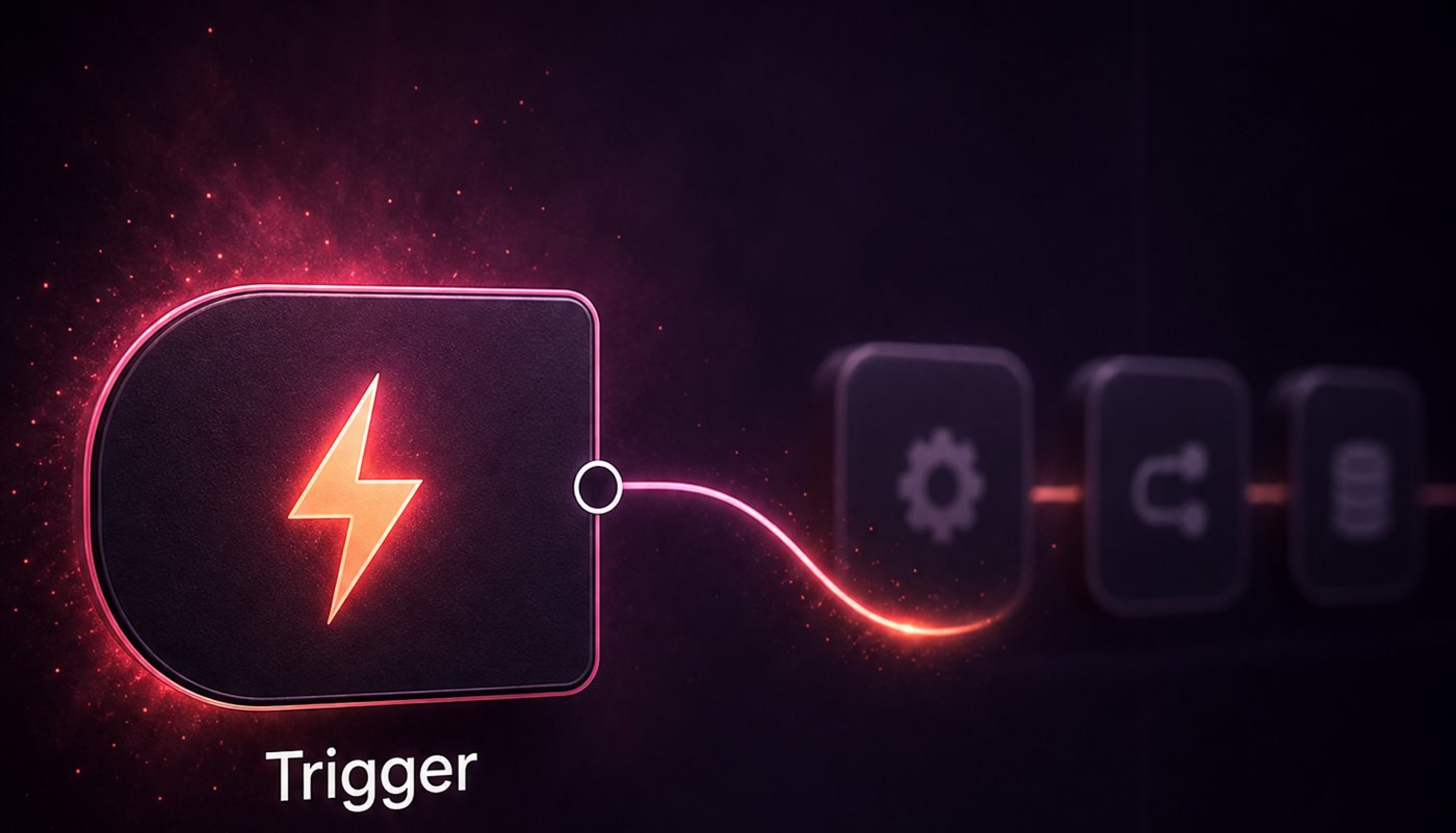
الفصل 6

المُحفِّزات: ما الذي يبدأ سير العمل؟

المُحفِّزات: يدوي وجدولة وخطّاف ويب

كل سير عمل يحتاج شرارة تبدأه





المُحفِّز: شرارة بدء سير العمل

المُحفِّز (Trigger) يحدد متى يعمل سير العمل

كل سير عمل إنتاجي يحتاج محفِّزا واحدا على الأقل

العقدة الأولى دائما، وله حافة منحنية مميزة

المُحفِّز اليدوي: زر للاختبار

لا إعدادات له، ولا يعمل تلقائيا

المُحفِّز اليدوي (Manual Trigger): تشغيل بنقرة
تضغط زر Execute Workflow فينطلق سير العمل

مثالي لتجربة سير العمل أثناء بنائه

مُحفِّز الجدولة: التشغيل بالوقت

مُحفِّز الجدولة (Schedule Trigger): يعمل بالوقت

اختر الفاصل: ثوانٍ، دقائق، ساعات، أيام، أسابيع

Custom (Cron) لجداول أدق بتعبير Cron

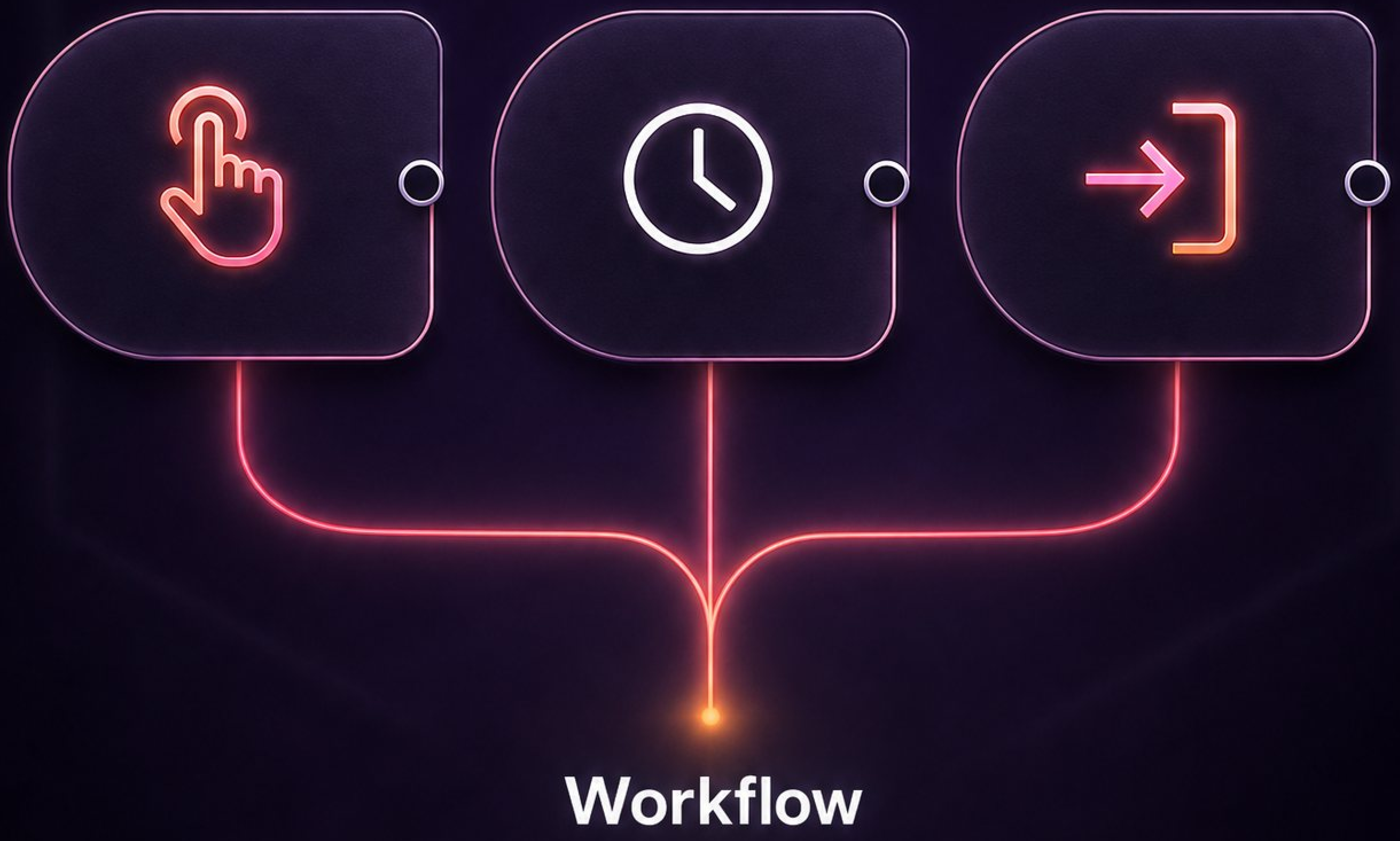
لا يعمل إلا بعد حفظ سير العمل ونشره

خطّاف الويب: استقبال طلب من الخارج

- خطّاف الويب (Webhook): رابط يستقبل البيانات
- يعمل عند وصول طلب HTTP من خدمة أخرى
- رابط اختبار للتجربة، ورابط إنتاج بعد النشر
- تختار الطريقة (POST/GET) والمسار والمصادقة

ماذا تعلَمت في هذا الفصل

كل سير عمل يحتاج محفّزا واحدا على الأقل



اليدوي: للاختبار بنقرة، بلا إعدادات

الجدولة: تشغيل متكرر بالوقت بعد النشر

خطّاف الويب: طلب HTTP وارد يبدأ التشغيل

الفصل 7

البيانات في n8n: العناصر و JSON

كل ما يمرّ بين العقد مصفوفة عناصر

كل شيء يسافر كمصفوفة عناصر

البيانات تتدفق كمصفوفة من العناصر (Items)

كل عنصر يساوي صفًا واحدا من البيانات

العقدة تستقبل عناصر وتُخرج عناصر

العنصر صندوق يحمل مفتاح json.

كل عنصر يحمل مفتاح json.

مفتاح binary اختياري للملفات والصور.

```
{ "json": { "name": "Mina",
"city": "Cairo" } }
```

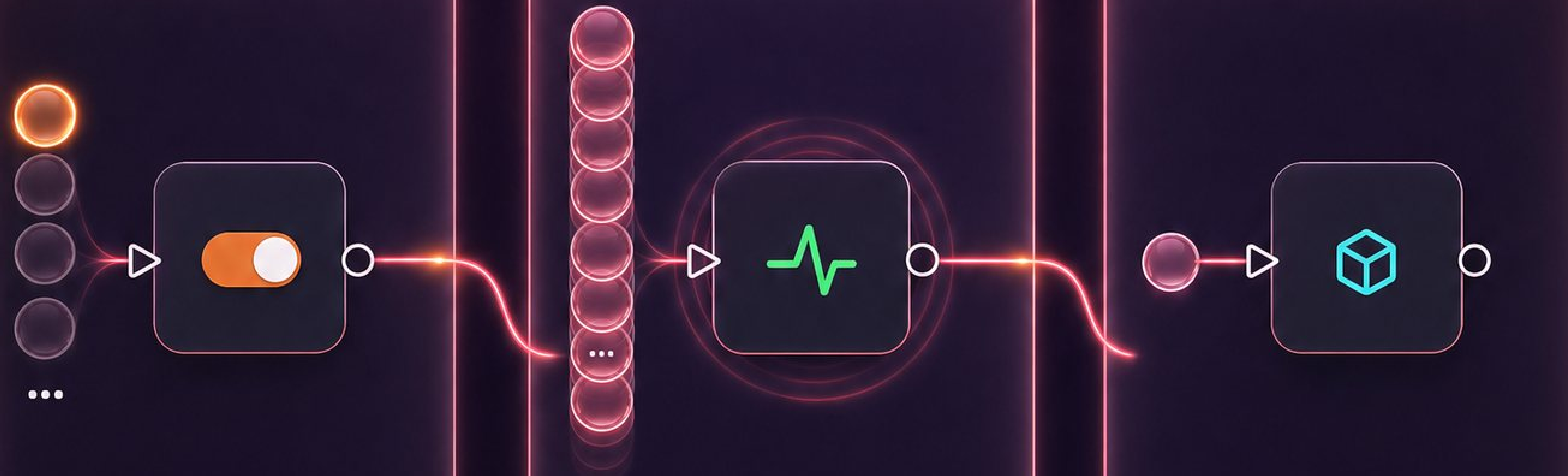
بنية العنصر الكاملة في n8n

```
[ { { "json": { "apple": "beets" },
  "binary": { "my-image": {
     "data": "...",
     "mimeType": "image/png",
     "fileName": "example.png"
 } } } ]
```

binary ، يحمل، data ، mimeType، fileName

قيمة data مُرمّزة بصيغة Base64

العقدة تعمل مرة لكل عنصر



مرة واحدة

عنصر واحد يدخل: العقدة تعمل مرة

مرة لكل عنصر

100 عنصر يدخل: العقدة تعمل 100 مرة

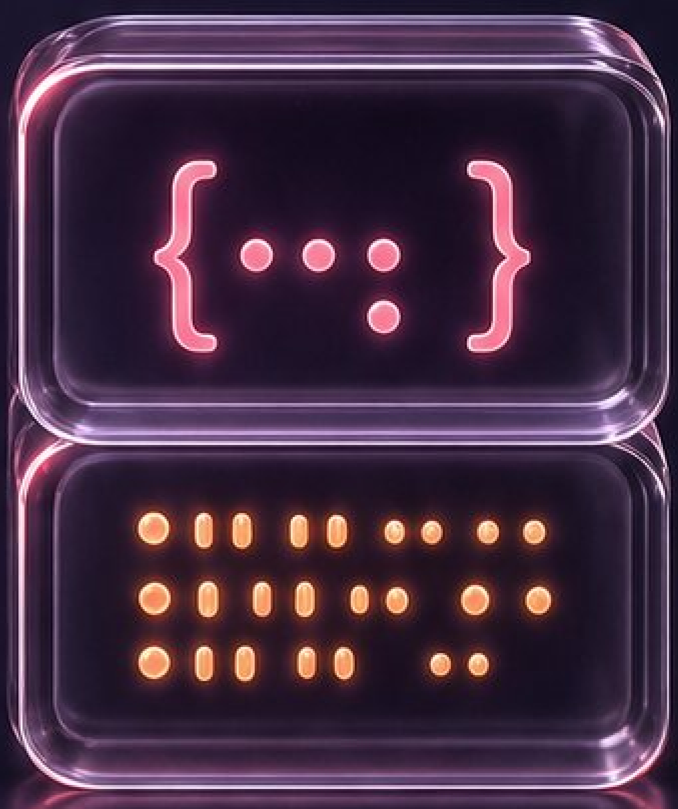
Execute Once

خيار Execute Once يعالج
العنصر الأول فقط

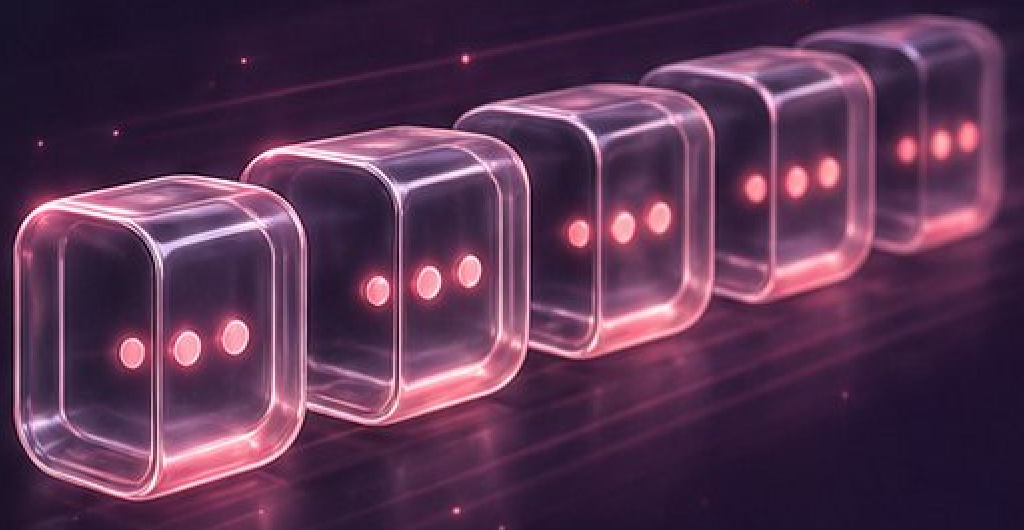
لا تحتاج كتابة حلقة تكرار بنفسك

ماذا تعلّمت في هذا الفصل

كل عنصر كائن json (وbinary اختياري)



البيانات تتدفق كمصفوفة من العناصر

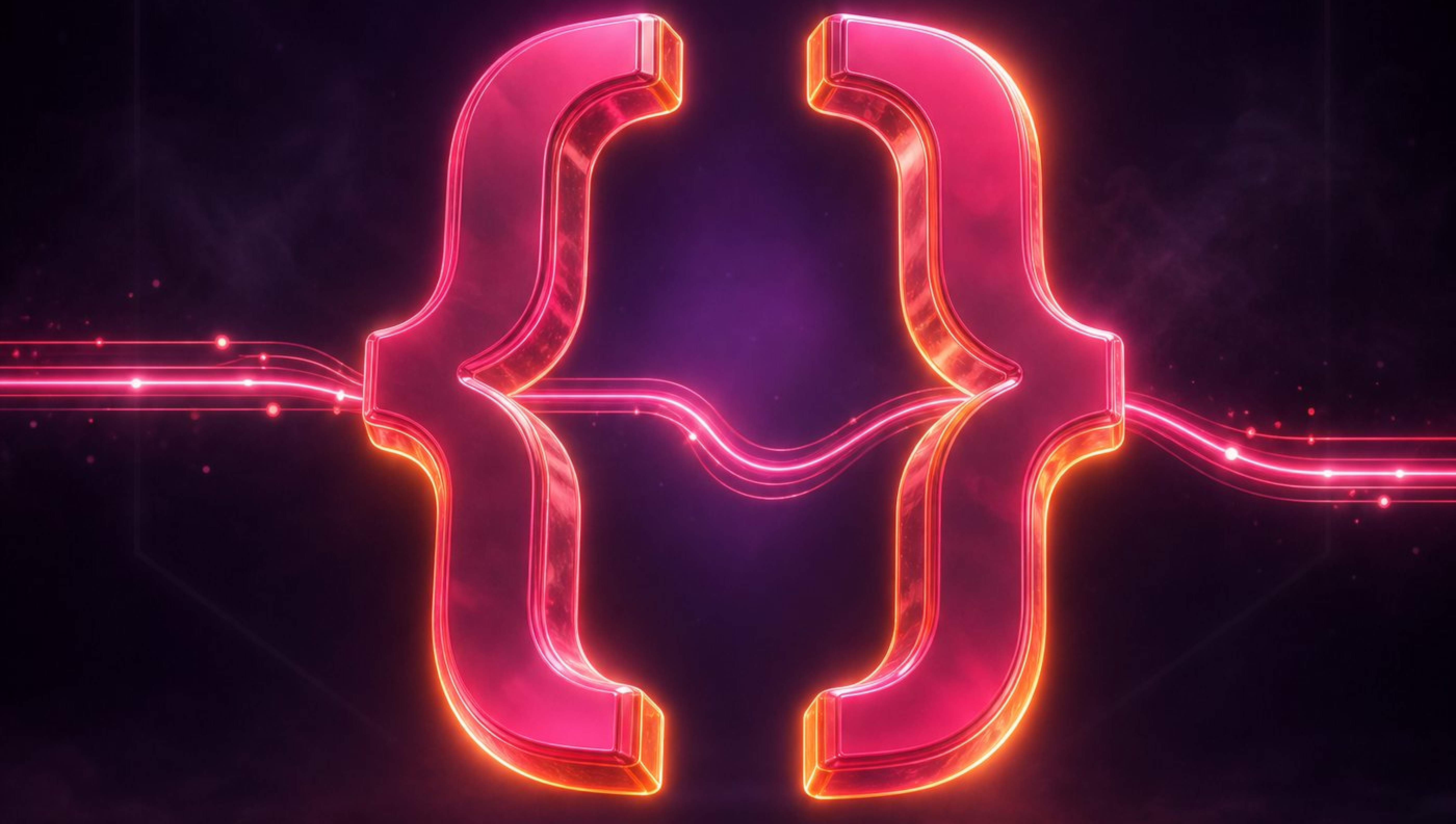


العقدة تعمل مرة لكل عنصر تلقائيا

بياناتك النصية تعيش تحت مفتاح json

الفصل 8

التعبيرات (Expressions)



اقرأ بيانات أي عقدة سابقة واستخدمها مباشرة

ما هو التعبير؟

التعبير: قيمة تُحسب وقت التشغيل

يُكتب داخل قوسين معقوفين مزدوجين

بداخله كود بلغة JavaScript

قيمة ثابتة مقابل تعبير

ثابتة

مرحبا بك

قيمة ثابتة: لا تتغير أبدا

مثال ثابت: مرحبا بك

تعبير

{{ $json.name }}

تعبير: يتغير مع كل عنصر

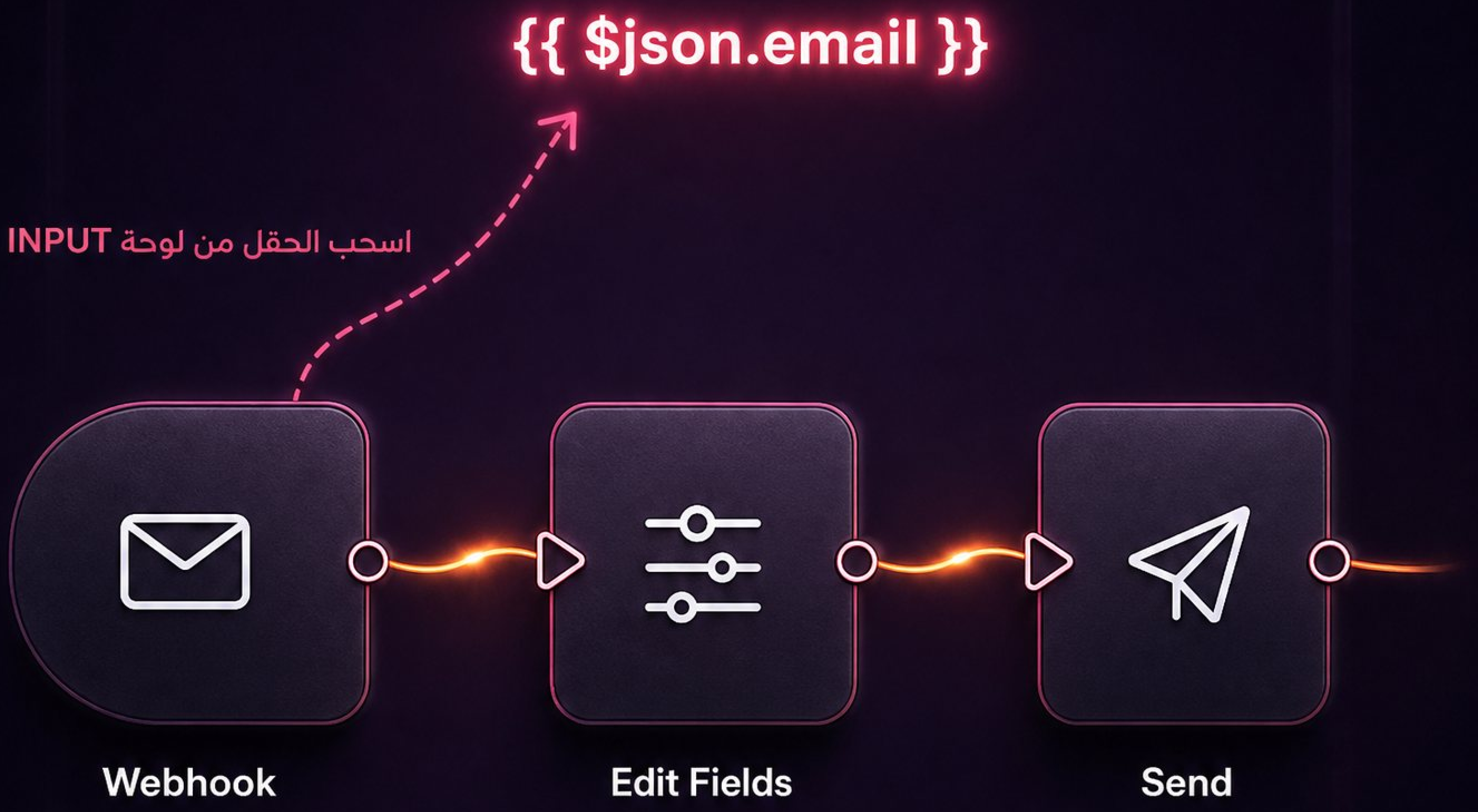
مثال تعبير: {{ $json.name }}

Fixed | Expression

بدّل من مفتاح **Fixed / Expression**

كيف تشير إلى البيانات

العنصر الحالي : {{ $json.email }}



عقدة سابقة : {{ $(ś'Webhook').item.json.id }}

التاريخ والوقت الآن : {{ $now }}

ماذا تعلّمت في هذا الفصل

اسحب وأفلِت ليكتب التعبير عنك

n8n

الفصل 9

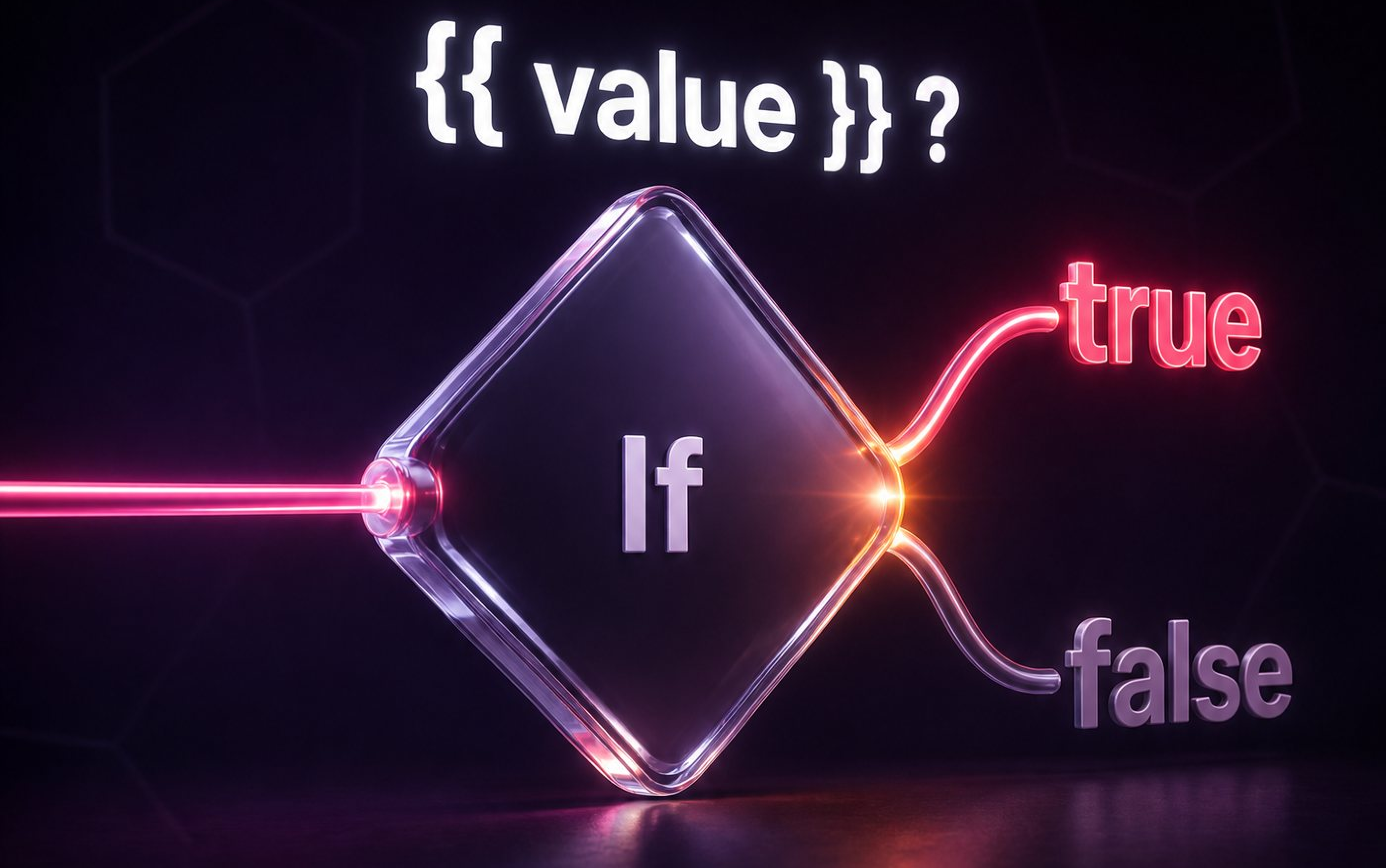
عقد المنطق:
و IN و Switch و IF

هنا يتعلّم سير عملك أن يقرّر بنفسه

IF | Switch | Filter

n8n

لماذا نحتاج إلى التفرّع؟



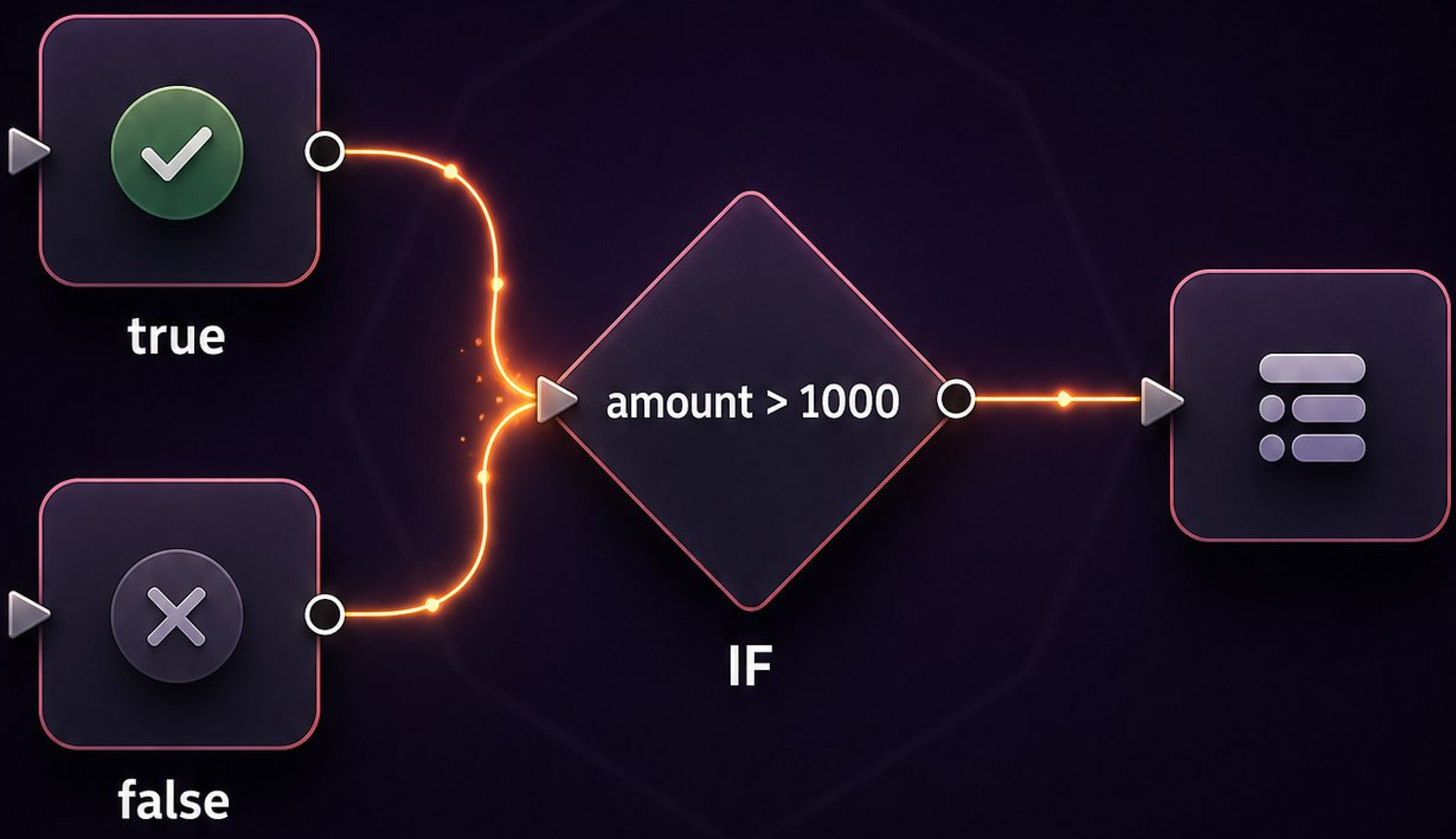
بدون منطق: كل عنصر يُعامَل بالطريقة نفسها

مع المنطق: لكل حالة مسارها الخاص

الشرط هو مقارنة قيمة بقيمة

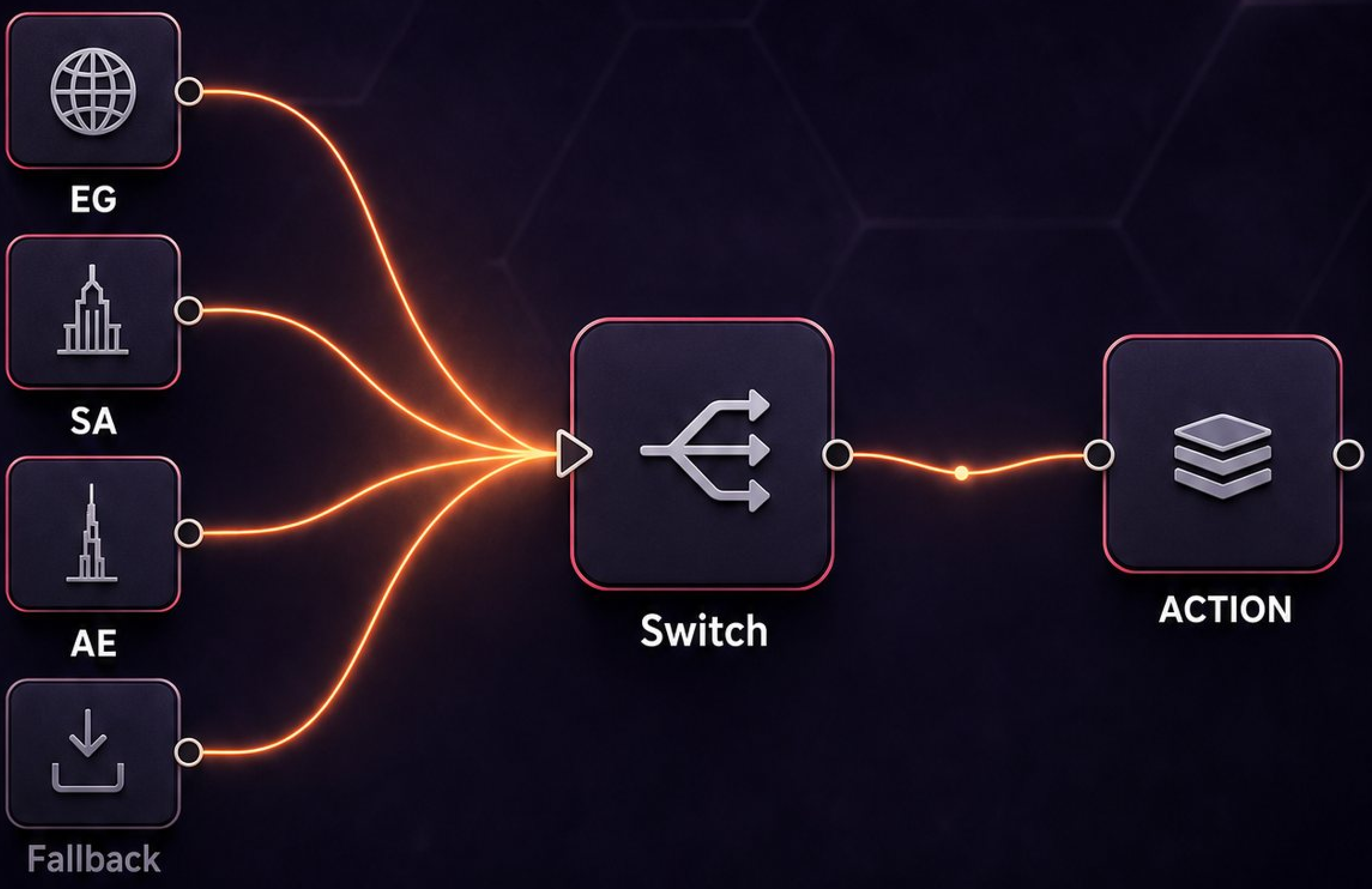
النتيجة: صحيح أو خطأ

عقدة الشرط (IF): مساران



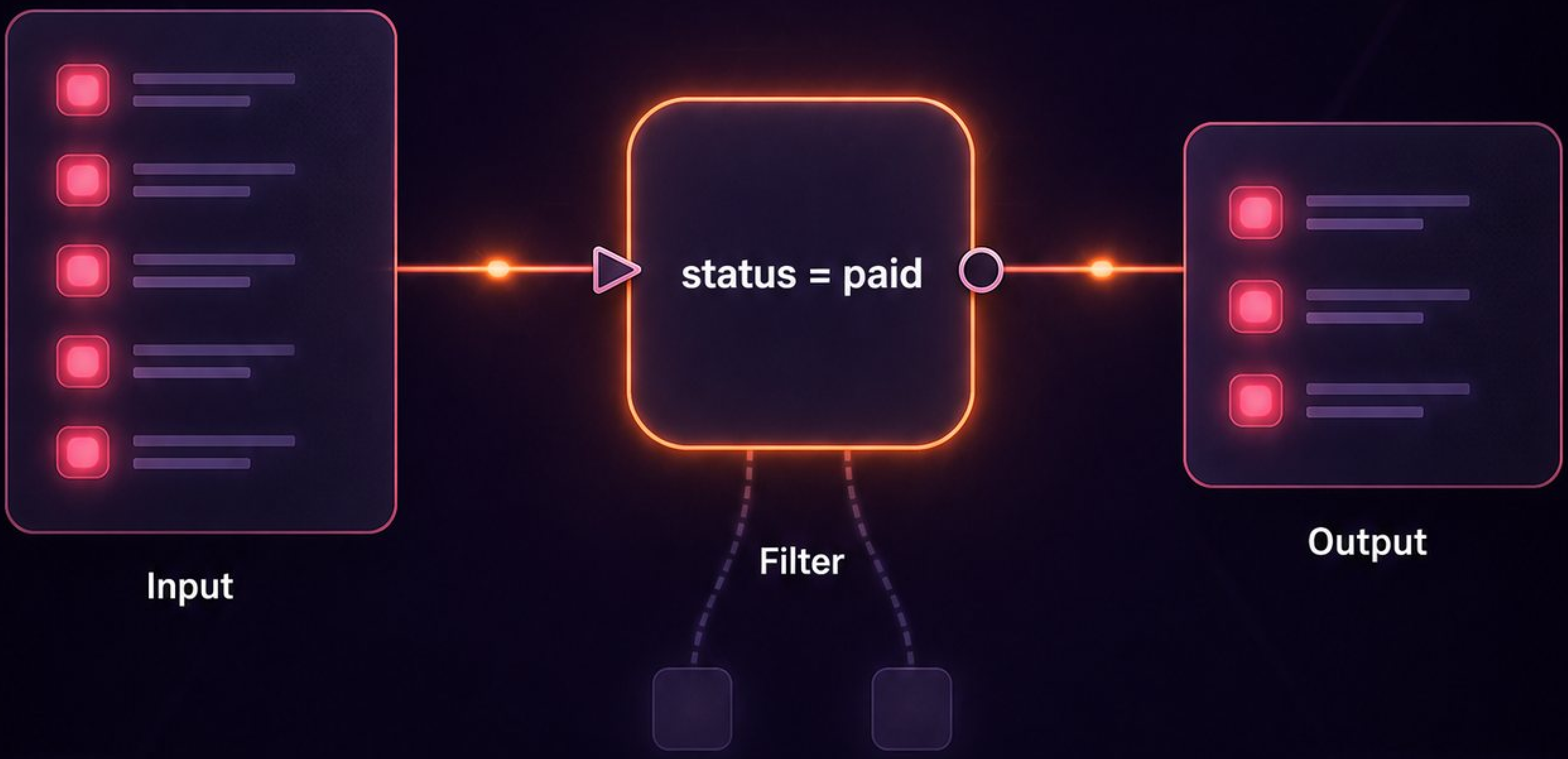
- مخرج true : العناصر التي تحقق الشرط
- مخرج false : العناصر التي تفشل فيه
- amount > 1000
- خيارات: Combinator و Ignore Case

عقدة التوجيه (Switch): مسارات متعددة



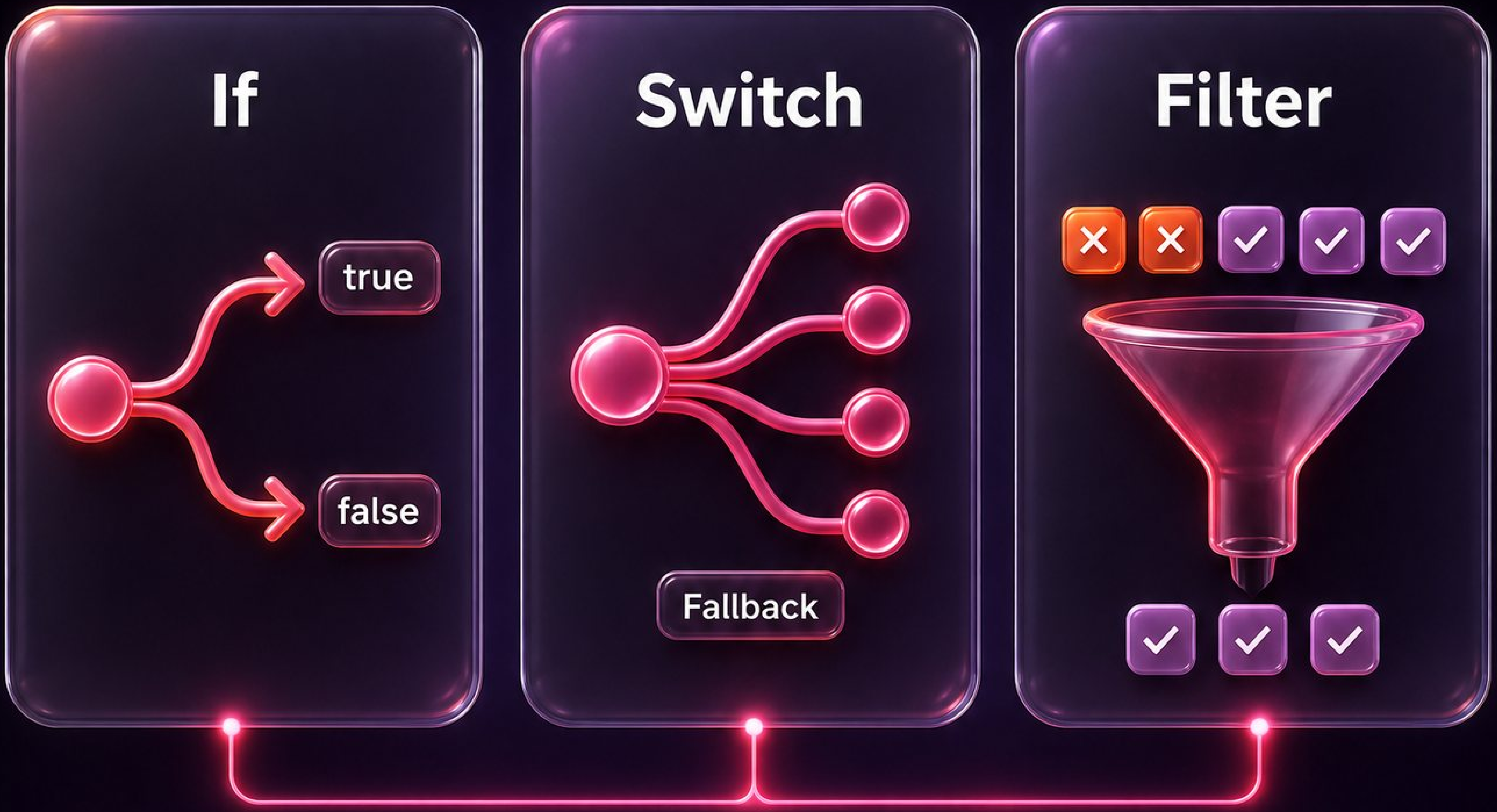
- قاعدة لكل مخرج بدل مسارين فقط
- مثال: توجيه حسب country
- Fallback Output لما لا يطابق شيء
- نمطان: Rules و Expression

عقدة التصفية (Filter): تنقية البيانات



- مخرج واحد فقط، لا تفرّع
- تمرّر ما يحقق الشرط فقط
- تستبعد الباقي بصمت
- OR و AND لا يُخلطان معا

ماذا تعلّمت في هذا الفصل



- مساران، true و false
- Fallback: عدة مسارات + Switch
- Filter: مخرج واحد، يُسقط الفاشل
- الشرط واحد: مقارنة قيمة بقيمة

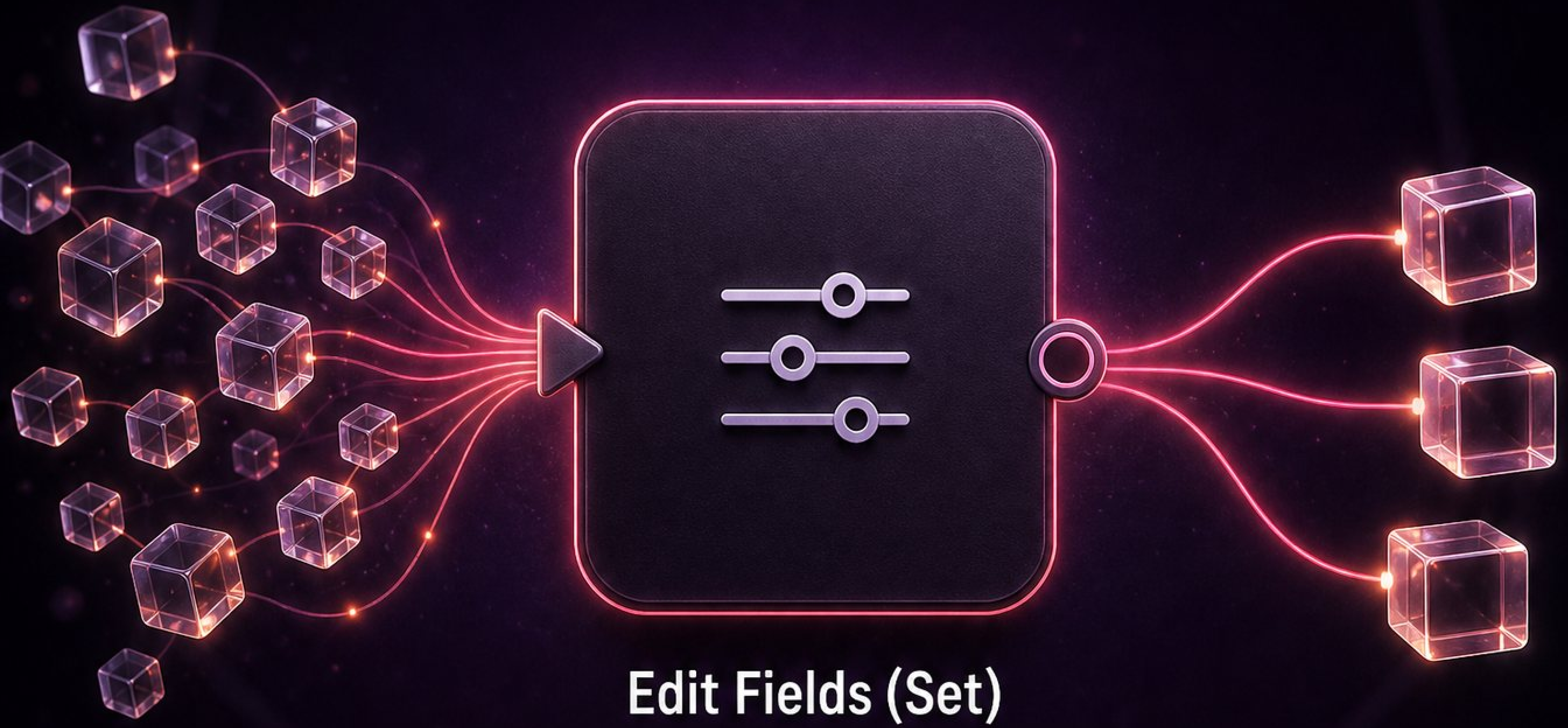
n8n

الفصل 10

تشكيل البيانات: تعديل الحقول والدمج والتكرار

نظِّف بياناتك، وادمج مسارين، وافهم التكرار

تعديل الحقول: اضبط بياناتك



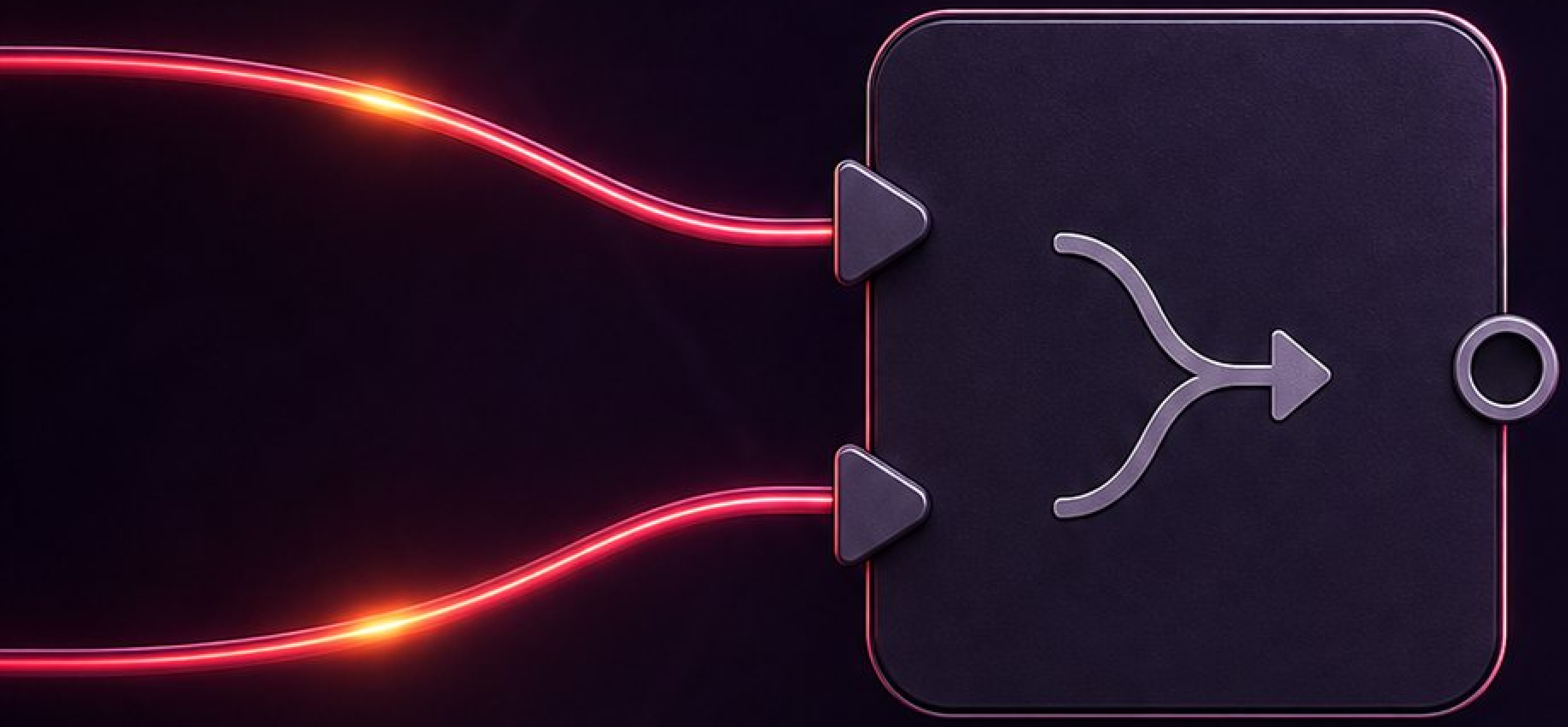

```
full_name = first + last
```

تعديل الحقول (Edit Fields / Set)

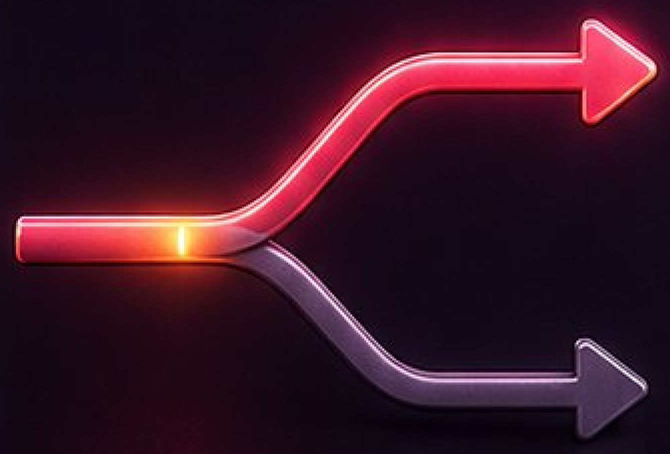
أضف حقولا جديدة، أو اكتب فوق القديمة

احتفظ بالحقول التي تختارها فقط

عقدة الدمج: اجمع مسارين



تعمل بعد توفر بيانات كل المسارات

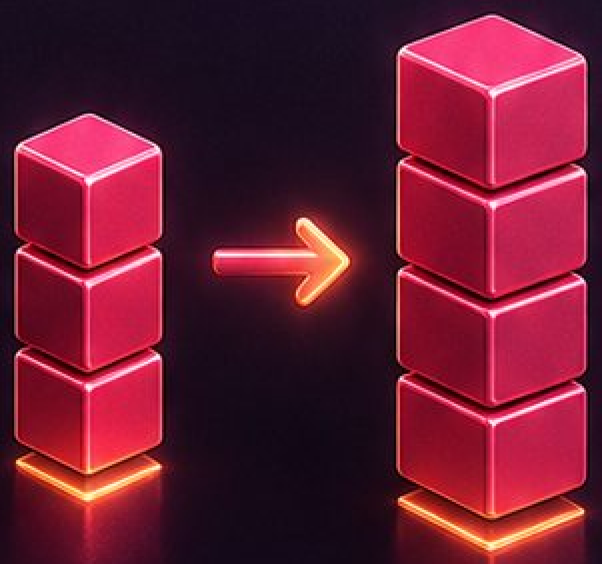


Choose Branch

Choose Branch: اختر
مخرج مسار واحد

Combine

Combine : طابق بحقل
أو بالموضع



Append

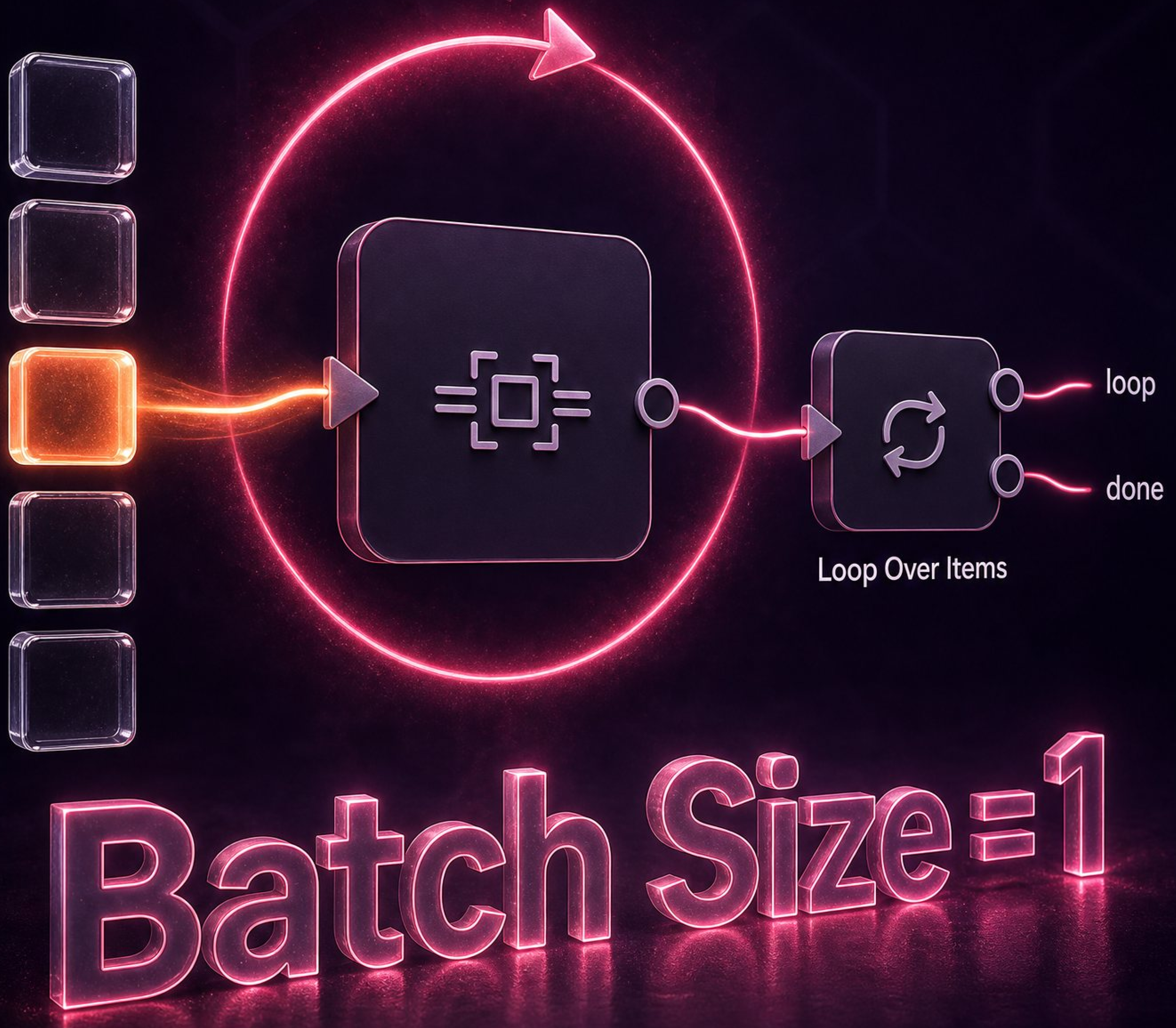
Append : ضع العناصر
فوق بعضها

العقد تعمل مرة لكل عنصر

غالبا تعمل العقدة مرة لكل عنصر

10 عناصر تعني 10 تشغيلات للعقدة

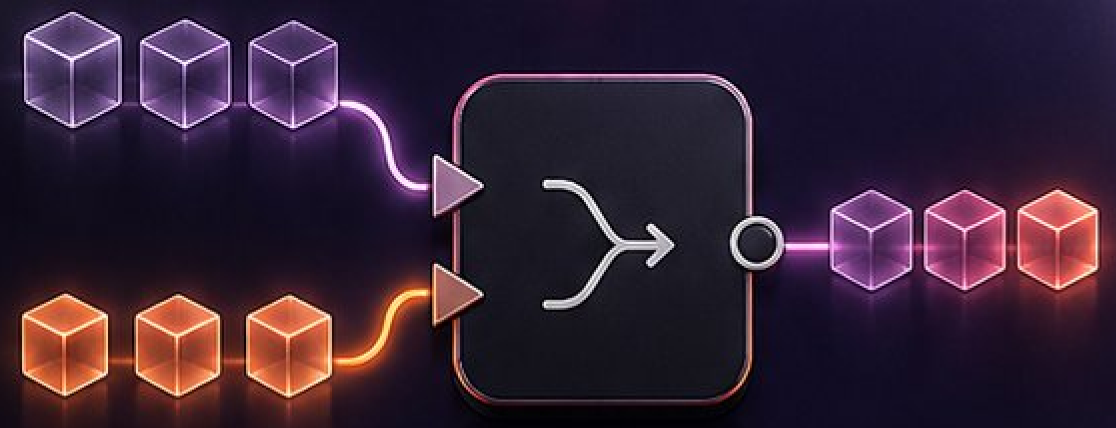
التكرار على العناصر (Loop Over Items): دفعات



ماذا تعلّمت في هذا الفصل

تعديل الحقول: أضف، أعد التسمية، نظّف

يبقي ما تختاره Keep Only Set Fields



Merge: Append أو Combine أو Choose Branch

العقدة تعمل مرة لكل عنصر تلقائيا

Loop Over Items للمعالجة على دفعات

الفصل 11

الاتصال بالخدمات: HTTP Request وعقد التطبيقات

كيف يخرج سير عملك إلى العالم

طريقتان للاتصال بالخدمات
طلب : HTTP: عامة لأي واجهة برمجية
عقدة التطبيق: جاهزة لخدمة بعينها
Gmail ، Google Sheets ، Telegram ، Slack
n8n
أكثر من 1000 تكامل جاهز في n8n

عقدة التطبيق: واجهة جاهزة

اختر المورد ثم العملية

Gmail: Message، Draft، Label، Thread

أرسل بريدا
دون كتابة كود

تتولى بيانات الاعتماد
المصادقة

طلب HTTP: المفتاح العام

- Method: GET للجلب، POST للإرسال
- URL: عنوان نقطة الواجهة
- Body و Headers و Query للمعاملات
- Authentication لبيانات الاعتماد

ماذا تعلّمت في هذا الفصل
REST
طلب HTTP: عامة لأي
واجهة برمجية REST
عقدة التطبيق:
جاهزة وأسهل، استخدمها أولاً
GET
POST
GET للجلب وPOST للإرسال
بيانات الاعتماد تتولى
المصادقة

الفصل 12

بيانات الاعتماد (Credentials) والأمان الأساسي

مفاتيحك السرية للاتصال بالخدمات، بأمان

ما هي بيانات الاعتماد؟

لماذا تُخزّن مشفّرة ومنفصلة؟

n8n يخزن كل بيانات الاعتماد مشفّرة

مفتاح تشفير يُنشأ تلقائيا في مجلد ~/.n8n

العقدَة تشير إلى السجل، لا تحتوي السر

تصدير سير العمل لا يصدّر الأسرار

مفتاح API مقابل OAuth
API
مفتاح API: تنسخ المفتاح
وتلصقه في الحقّل
sk_live_••••
OAuth
أوّث (OAuth): تضغط زر
تسجيل الدخول للموافقة
Connect
كلاّهما يُحفظ كبيانات اعتماد مشفّرة
أوث لا يكشف كلمة مرورك للخدمة

كيف تربط بيانات الاعتماد بعقدة

1 افتح العقدة واختر Create new credential

2 أدخل المفتاح أو سجِّل الدخول عبر أوث (OAuth)

3 n8n يختبر الاتصال تلقائيا عند الحفظ

4 أعِد استخدام السجل نفسه في أي عقدة لاحقا

عادات أمان أساسية

لا تلصق الأسرار في حقول
نصية أو تعابير ظاهرة

سمّ كل سجل باسم
واضح يدل على غرضه

احذف بيانات الاعتماد التي
لم تعد تستخدمها

ماذا تعلّمت في هذا الفصل

n8n يخزّنها مشفّرة
ويختبرها عند الحفظ

بيانات الاعتماد سجل سري
منفصل عن سير العمل

القاعدة الذهبية: لا تلصق
أسرارك في الحقّول

أوث للربط بنقرة،
ومفتاح API باللصق

الفصل 13:
معالجة الأخطاء الأساسية

معالجة الأخطاء (Error Handling) الأساسية

اجعل سير عملك يفشل بهدوء، لا في صمت

لماذا تفشل سيور العمل؟

بيانات سيئة: حقل ناقص أو نوع خاطئ

مصادقة: بيانات اعتماد منتهية أو خاطئة

حدود المعدّل: طلبات أكثر مما تسمح الخدمة

الفشل الصامت يضرّ أكثر من الظاهر

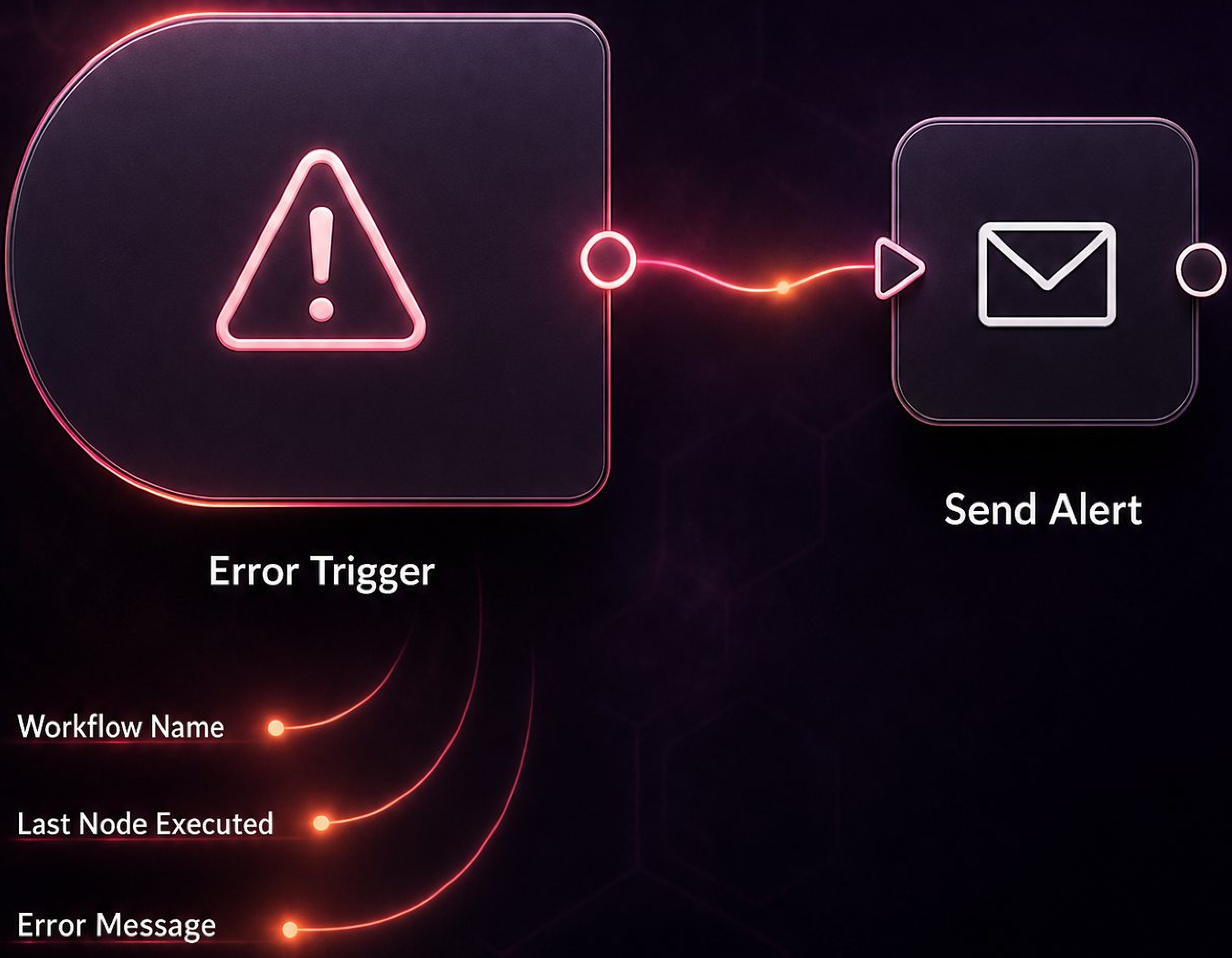
مُحفِّز الأخطاء وسير عمل الأخطاء

مُحفِّز الأخطاء (Error Trigger): يبدأ عند الفشل

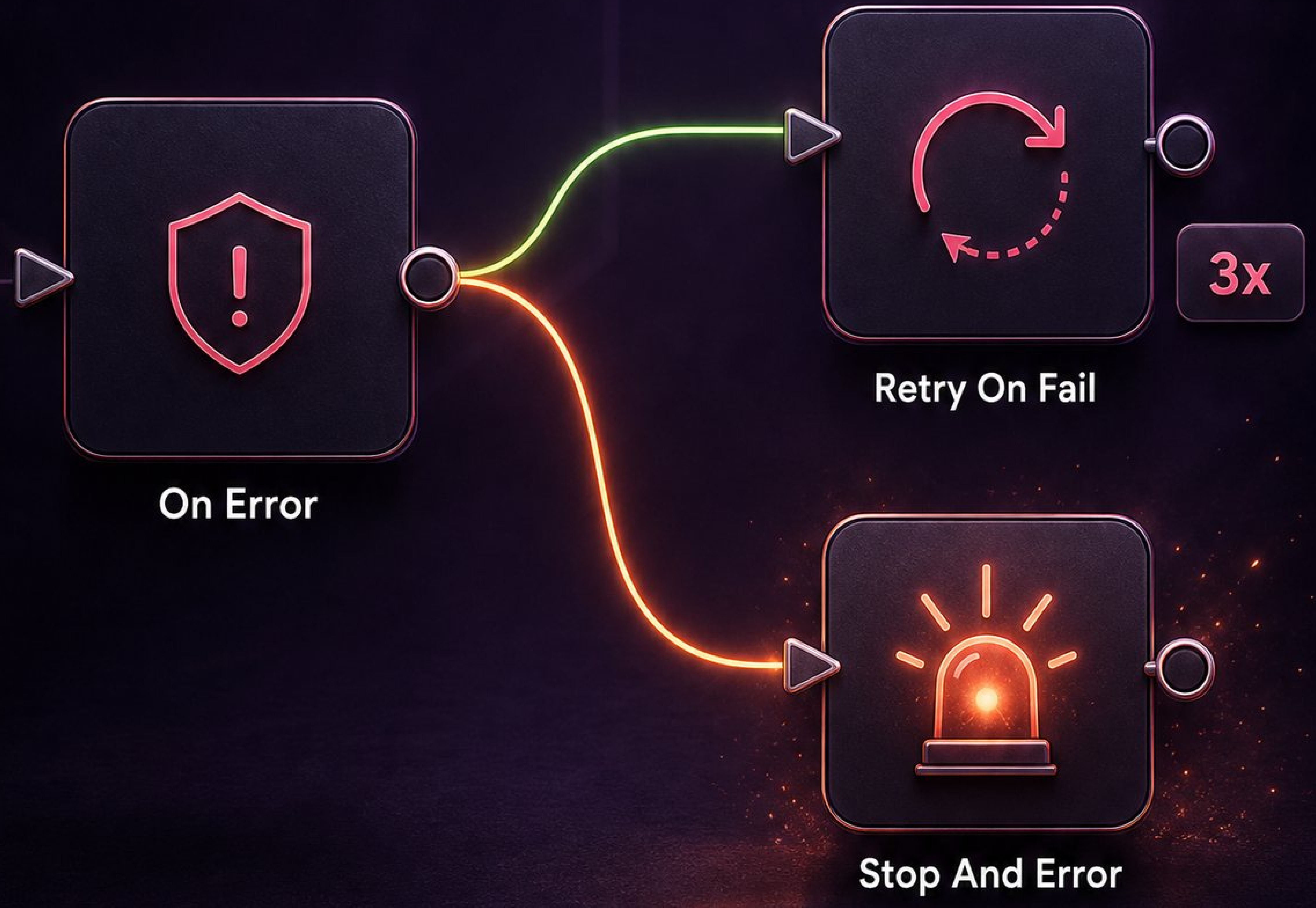
اربطه من إعدادات سير العمل

يصلك: اسم سير العمل، آخر عقدة، نص الخطأ

يعمل في التشغيل التلقائي فقط، لا الاختبار



إعدادات العقدة: تجاوز أم توقّف؟



On Error: Stop Workflow (الافتراضي)

Continue / Continue (using error output): تجاوز

Retry On Fail: مع Max. Tries و Wait Between Tries

Stop And Error: أوقف التشغيل عمدا برسالتك

n8n

الفصل 14

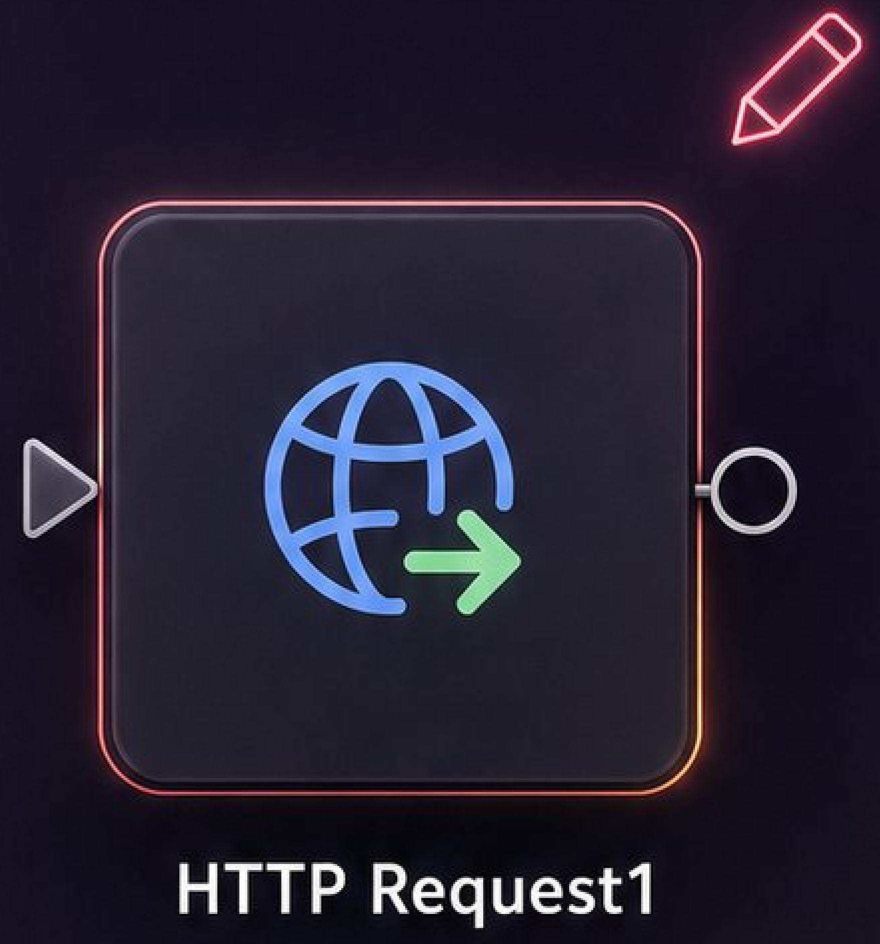
عادات صغيرة تبقي عملك مرتبا

سمِّ، وثِّق، صغْر، واختبر

سمِّ ووثِّق ما تبنيه

1. أعد تسمية كل عقدة باسم واضح

لا تترك ‘HTTP Request1’،
اكتب ‘جلب الطلبات‘



2. اشرح المنطق بملاحظة لاصقة

الملاحظة (Sticky Note) تدعم
Markdown والألوان

تذكّر: تكتب لنفسك بعد شهر، لا للآلة

صغِّر وقسِّم بدل التعقيد

اجعل كل سير عمل يؤدي مهمة واحدة واضحة

المنطق المتكرر؟ انقله إلى سير عمل فرعي

عقدة 'Execute Sub-workflow' تستدعيه

نظّم سيور العمل في مجلدات (Folders) ووسوم (Tags)

اختبر بذكاء، ولا تُفرط

ثبّت مخرجات عقدة للاختبار دون إعادة الجلب

البيانات المثّبّتة (Pinned data) للتشغيل اليدوي فقط

اختبر مسارا واحدا قبل أن تبنـي الباقي

ليس كل شيء يستحق الأتمتة

ماذا تعلّمت في هذا الفصل

سيور عمل صغيرة،
والمتكرر في سير عمل فرعي

أسماء واضحة وملاحظات
لاصقة تشرح المنطق

نظّم بالمجلدات والوسوم،
ولا تُفرط في الأتمتة

البيانات المثبّتة
تسرّع الاختبار اليدوي

n8n

الفصل 15

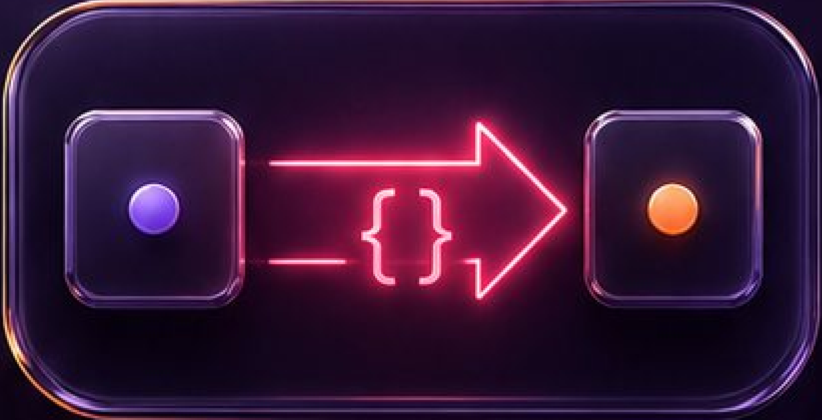
ثلاثة أمثلة عملية كاملة

من المُحفِّز إلى آخر عقدة، خطوة بخطوة

كيف نقرأ كل مثال

1) المُحفّز: متى يبدأ التشغيل

2) العُقّد بالترتيب



3) ما الذي يتدفق بين العُقّد

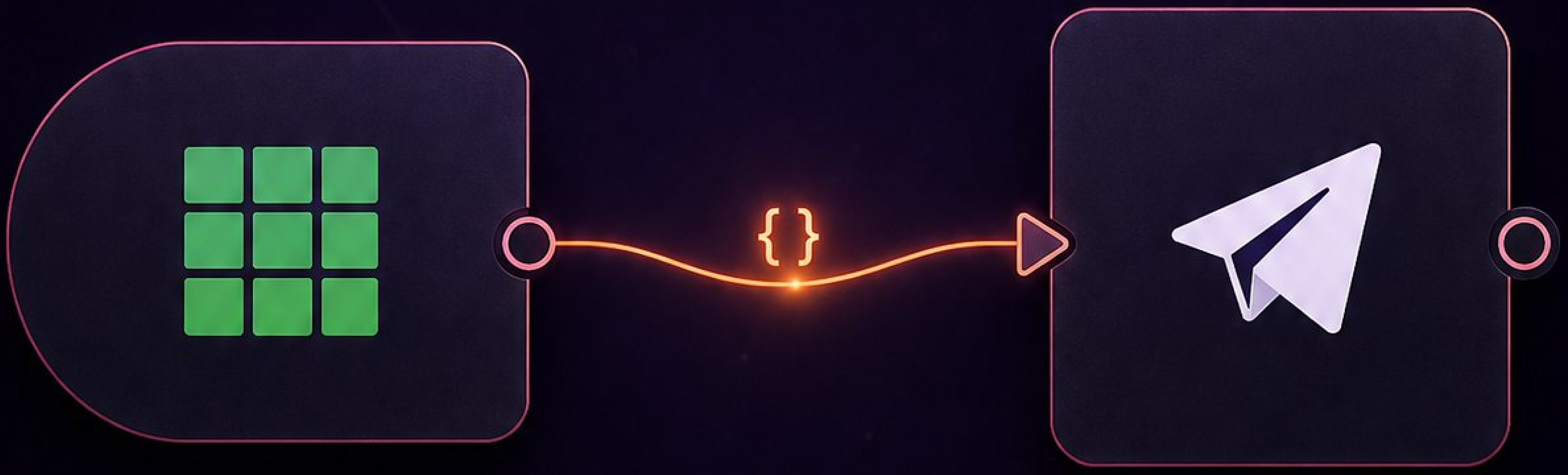
ابنِ كل مثال بنفسك خطوة بخطوة

المثال ١: إشعار من Google Sheets

صف جديد ← إشعار فوري

Google Sheets Trigger ← Telegram

المثال 1: العُقد والتدفق



1) Google Sheets Trigger: Row Added

2) Telegram ← Send Message

Text: {{ $json.name }} طلب جديد

Chat ID + Access Token من BotFather

المثال 2: نشرة يومية من RSS

كل صباح ← ملخص أخبار بالبريد

Trigger at Hour: 8

Schedule Trigger ← RSS Read ← Send Email

المثال 2: العُمَد والتدفق

Subject + قائمة {{ $json.title }}

المثال 3: نموذج ← Sheet ← تأكيد

إرسال نموذج ← تسجيل ← رسالة تأكيد

Webhook ← Google Sheets ← Send Email

Append Row ثم تأكيد للعميل

المثال 3: العُقد والتدفق

To: {{ $json.body.email }}

ماذا تعلّمت في هذا الفصل

البيانات تتدفق كعناصر
عبر الوصلات

كل سير عمل: مُحفّز
ثم عُقَد متسلسلة

غيّر مُحفّزا أو عقّدة
وستبني أي شيء

التعبيرات {{ }} تنقل
القيم بين العُقّد

n8n

16

الفصل 16

الختام بالذكاء الاصطناعي والخطوات التالية

بريد يصل، الذكاء الاصطناعي يلخّصه، يصلك على Telegram

ثم: القوالب والمجتمع وما ينتظرك في الإصدار 2

الفكرة: ثلاث خطوات بسيطة

بريد يصل ← يُلخّص بالذكاء الاصطناعي ← يصلك على Telegram

مُحفِّز، ثم عقدة ذكاء اصطناعي، ثم إجراء

نفس المفاهيم التي تعلّمتها، بقدرة جديدة

عقدة الذكاء الاصطناعي: Basic LLM Chain

النموذج اللغـوي الكبير (LLM): يفهم النص ويولّده

Basic LLM Chain: نداء واحد، مهمة واحدة

اربط به نموذج محادثة مثل OpenAI Chat Model

اطلب منه: «لخّص هذا البريد في 3 أسطر»

حين تحتاج أكثر: وكيل الذكاء الاصطناعي

وكيل الذكاء الاصطناعي (AI Agent): يقرر ويستخدم أدوات

نموذج محادثة + ذاكرة + أدوات

يعمل دائما الآن كوكيل أدوات (Tools Agent)

للتلخيص يكفيك Basic LLM Chain

ماذا تعلّمت في هذا الفصل

لمهمة Basic LLM Chain
واحدة، AI Agent للقرار والأدوات

بنيت سير عمل ذكيا:
بريد ← تلخيص ← Telegram

أكمل عبر القوالب
والمجتمع والإصدار 2

كل نموذج محادثة
يحتاج بيانات اعتماد

مسرد المصطلحات (1/2)

n8n Cloud	n8n السحابية
Community edition	نسخة المجتمع
fair-code	الكود العادل
self-hosted	استضافة ذاتية
no-code / low-code	بدون كود / بكود قليل
task	مهمة (وحدة فوترة Zapier)
Workflow	سير عمل
Sub-workflow	سير عمل فرعي
Template	قالب
Execution	تشغيل
Publish workflow	نشر سير العمل
Canvas	اللوحة
Nodes panel	لوحة العقد
Sticky Note	ملاحظة لاصقة
Folder	مجلد
Tag	وسم
Node	عقدة
Core node	عقدة أساسية
App node	عقدة تطبيق
Cluster nodes	العُقد العنقودية
Connection	وصلة
Trigger	مُحفّز
Manual Trigger	مُحفّز يدوي
Schedule Trigger	مُحفّز جدولة
Webhook	خطّاف ويب
Respond to Webhook	الرد على خطّاف الويب
Error Trigger	مُحفّز الأخطاء
Action	إجراء
Resource (in app nodes)	المورد
Operation (in app nodes)	العملية
Item	عنصر
Array	مصفوفة
JSON	جيسون
Base64	بيس64
Pinned data	بيانات مثبّتة
Execute Once	التنفيذ مرة واحدة

مسرد المصطلحات (2/2)

Expression	تعبير
IF	عقدة الشرط
Switch	عقدة التوجيه
Filter	عقدة التصفية
Merge	عقدة الدمج
Edit Fields (Set)	تعديل الحقول
Loop Over Items (Split in Batches)	التكرار على العناصر
Code node	عقدة الكود
Stop And Error	عقدة الإيقاف بخطأ
HTTP Request	طلب HTTP
Method (HTTP)	الطريقة
API	واجهة برمجية
REST API	واجهة REST
API key	مفتاح الواجهة البرمجية
Credential	بيانات اعتماد
OAuth	أوث
Encryption key	مفتاح التشفير
On Error	عند الخطأ (إعداد العقدة)
Retry On Fail	إعادة المحاولة عند الفشل
Error workflow	سير عمل الأخطاء
Google Sheets Trigger	مُحفّز Google Sheets
Gmail Trigger	مُحفّز Gmail
RSS Read	قراءة RSS
Send Email	إرسال بريد
Telegram (node)	عقدة Telegram
AI Agent	وكيل ذكاء اصطناعي
Tools Agent	وكيل الأدوات
LLM	نموذج لغوي كبير
Chat Model	نموذج محادثة
Basic LLM Chain	سلسلة النموذج اللغوي الأساسية
Memory (Simple Memory)	ذاكرة بسيطة

تابع www.n8nbilarabi.com

مينا سعد