



جامعة طرابلس كلية تقنية المعلومات



Advanced Databases قواعد البيانات المتقدمة ITSE312

أستاذ المادة - حسن علي حسن

h.ebrahem@uot.edu.ly

المحاضرة التاسعة - القادح

TRIGGER





مواضيع المحاضرة التاسعة

- ▶ قادح TRIGGER
- ▶ إنشاء قادح CREATE TRIGGER
- ▶ استخدام معامل القيمة الجديدة NEW ومعامل القيمة القديمة OLD
- ▶ مزايا القادح Advantages TRIGGER
- ▶ عيوب القادح Disadvantages TRIGGER



القادح TRIGGER

- ▶ السجلات في الجداول تحتاج دائما إلى عملية تحديث في بياناتها باستخدام أوامر التحديث (الإدخال، التعديل، الحذف)، وقد نحتاج إلى معرفة التحديثات التي تمت على البيانات داخل الجداول، يمكن معرفة ذلك باستخدام القادح Trigger.
- ▶ أحيانا نحتاج إلى معرفة التغييرات التي تمت على السجلات داخل جداول قاعدة البيانات، عملية متابعة التغييرات التي تحدث تحتاج إلى نظام آلي يشتغل دائما داخل قاعدة البيانات، يمكن القيام بذلك عن طريق القادح Trigger. يعمل القادح بشكل آلي بدون التدخل المباشر من المستخدم، تدعم لغة SQL القوادح Triggers والتي يتم تنفيذها بشكل تلقائي عند حدوث حدث Action معين على بيانات الجداول.

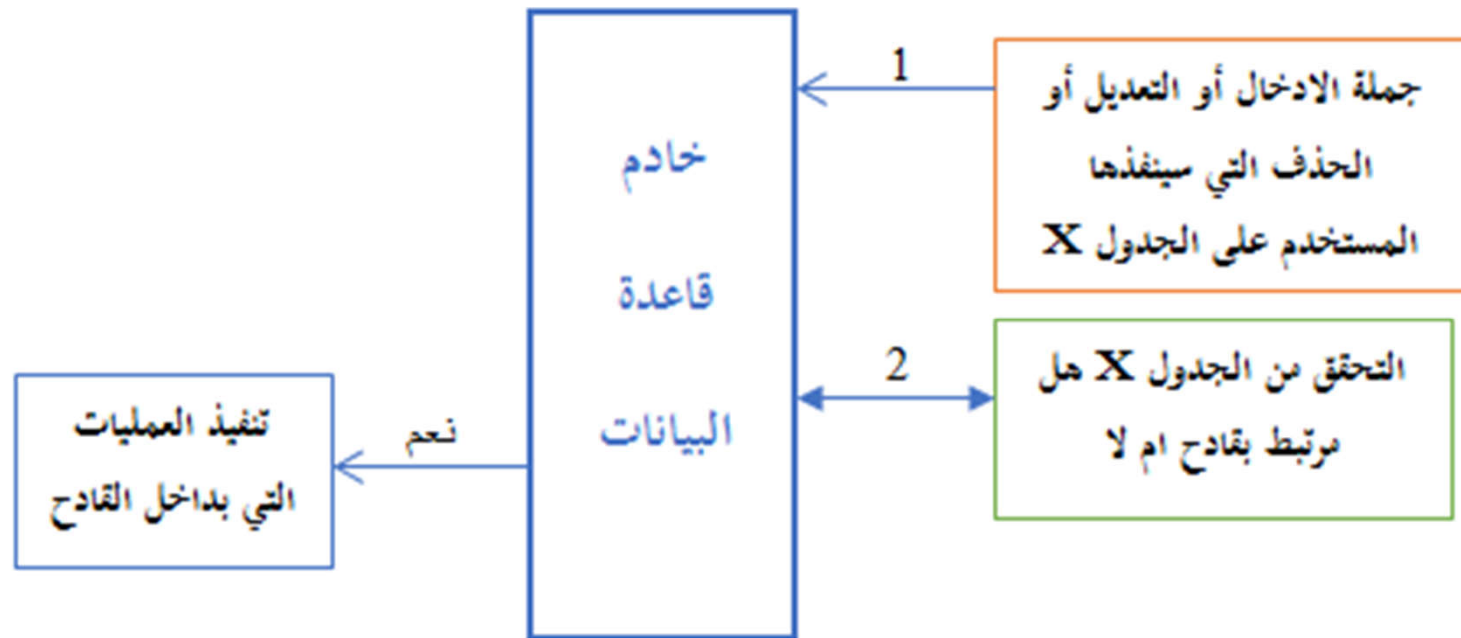
القادح TRIGGER

▶ القادح ويسمى النابض أو المحفز وهو عبارة عن حدث مخزن في قاعدة البيانات ومرتبطة بجدول، يتم تنفيذه بشكل آلي عند حدوث أي نشاط معين مثل إدخال Insert، تعديل Update أو حذف Delete على الجدول.

▶ القادح **Trigger** يشابه الإجراء المخزن Stored Procedure إلا أنه لا يتم تنفيذه باستدعاء اسمه، إنما يتم تنفيذه بشكل آلي في حالة إجراء عملية Insert, Update or Delete على جدول معين. على سبيل المثال، إذا تم ربط القادح مع جدول X عند حدوث عملية إدخال في الجدول X، فإن القادح يشتغل آليا في حالة قيام المستخدم بإدخال بيانات في الجدول.

القادح TRIGGER

خطوات تشغيل القادح



إنشاء قاذح CREATE TRIGGER

▶ تختلف تركيبة القاذح على حسب نظام DBMS، ينصح بالاطلاع على التوثيق الخاص بنظام DBMS، على سبيل المثال، نظام MySQL قبل الاصدار 5.1 لا يدعم القوادح.

▶ الشكل يبين تركيبة القاذح في نظام MySQL.

```
DELIMITER $$
```

```
CREATE TRIGGER اسم الجدول ON حدث التنفيذ وقت التنفيذ اسم القاذح  
FOR EACH ROW
```

```
BEGIN
```

```
    ; جملة الإدخال أو التعديل أو الحذف
```

```
END $$
```

```
DELIMITER ;
```

استخدام معامل القيمة الجديدة **NEW** ومعامل القيمة القديمة **OLD**

يوجد معاملان يستخدمهما القادح للوصول إلى قيم الخصائص قبل وبعد التحديث في الجدول وهما معامل القيمة الجديدة **NEW** ومعامل القيمة القديمة **OLD**.

▶ القادح يستطيع الوصول إلى كافة البيانات الجديدة التي تم إدخالها بواسطة أمر **INSERT** أو أمر **UPDATE** عن طريق معامل **NEW**.

▶ القادح يستطيع الوصول للبيانات القديمة عن طريق معامل **OLD** التي تم تعديلها بأمر **UPDATE** أو التي تم حذفها بأمر **DELETE**، بمعنى الوصول للبيانات قبل التعديل وقبل الحذف.



مثال 1

▶ إنشاء قاذح باسم **قبل_الإدخال** يقوم بعملية التعديل UPDATE في جدول الملكية على خاصية الإيجار_الشهري بطرح 100 منها لجميع الملكيات، وذلك قبل BEFORE القيام بعملية الإدخال INSERT في نفس جدول الملكية.

```
DELIMITER $$  
CREATE TRIGGER قبل_الإدخال BEFORE INSERT ON الملكية  
FOR EACH ROW  
BEGIN  
    UPDATE الملكية SET الإيجار_الشهري = الإيجار_الشهري - 100 ;  
END $$  
DELIMITER ;
```


يتبع مثال 1 خطوات تشغيل القادح باسم (قبل_الادخال)

DELIMITER \$\$

CREATE TRIGGER *الملكية* BEFORE INSERT ON *الملكية*

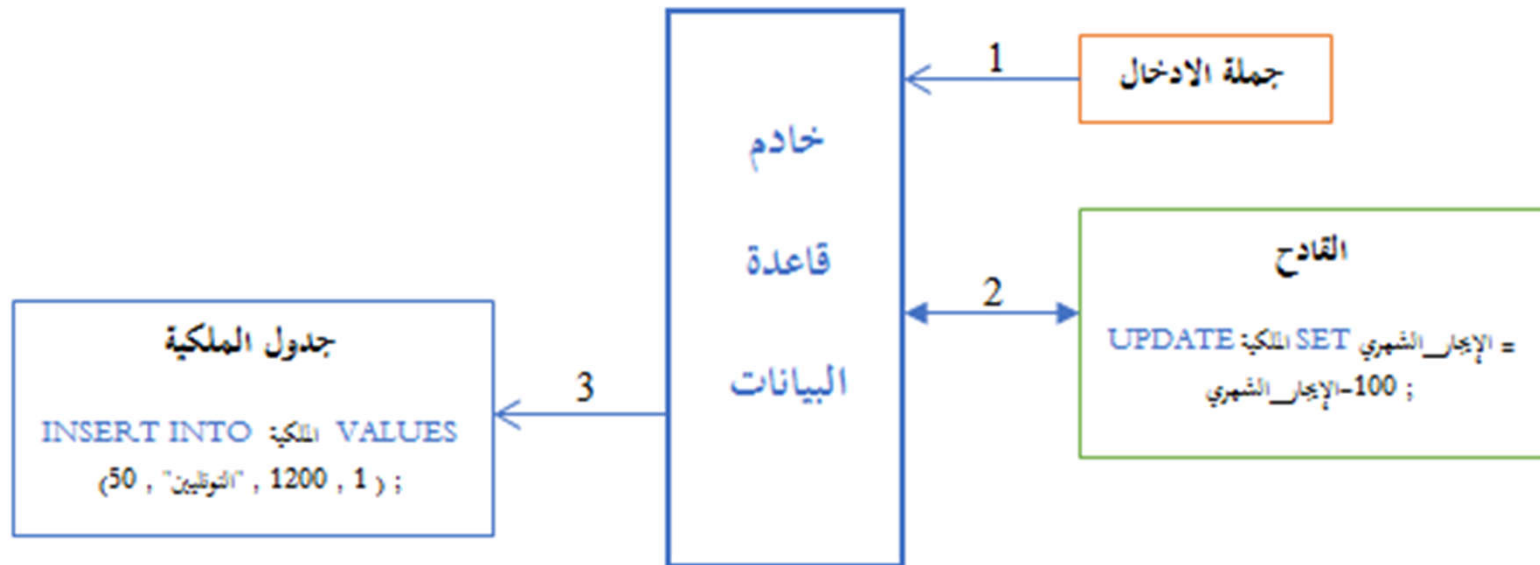
FOR EACH ROW

BEGIN

UPDATE *الملكية* SET *الإيجار_الشهري* = *الإيجار_الشهري* - 100 ;

END \$\$

DELIMITER ;



مثال 1

جدول الملكية

رقم_مالك_العقار	الإيجار_الشهري	عنوان_الملكية	رقم_الملكية
1	1500	السراج	10
2	1000	بن عاشور	20
2	2000	قرقارش	30
3	1500	زناته	40

عندما يقوم المستخدم بتنفيذ عملية إدخال البيانات في جدول الملكية كما في الشكل التالي:

```
INSERT INTO الملكية VALUES (50 , "النوفلين" , 1200 , 1 ) ;
```

فإن البيانات التي في جدول الملكية تتغير كما في الجدول التالي:

رقم_مالك_العقار	الإيجار_الشهري	عنوان_الملكية	رقم_الملكية
1	1400	السراج	10
2	900	بن عاشور	20
2	1900	قرقارش	30
3	1400	زناته	40
1	1200	النوفلين	50

مثال 2

- ▶ لزيادة توضيح عمل القادح، سيتم إنشاء جدول جديد باسم تعديل_الملكية، في هذا الجدول سيقوم القادح بتخزين بيانات من بعض الخصائص في جدول الملكية التي تم تعديل بياناتها من قبل المستخدم.
- ▶ **مثال:** إنشاء جدول جديد باسم تعديل_الملكية يتم فيه تخزين التعديلات التي تحدث في جدول الملكية باستخدام القادح، أنظر الشكل.

```
CREATE TABLE تعديل_الملكية (  
    رقم_تسلسل INT(11) AUTO_INCREMENT PRIMARY KEY,  
    رقم_الملكية1 INT(2) NOT NULL, رقم_مالك_العقار1 INT(2) NOT NULL,  
    التاريخ DATE DEFAULT NULL,  
    الإيجار_القديم INT(6) DEFAULT NULL,  
    الإيجار_الجديد INT(6) DEFAULT NULL ) ;
```

مثال 2

▶ بعد إنشاء الجدول نبدأ بإنشاء قادح باسم بعد_التعديل يقوم بعملية الإدخال
INSERT في جدول تعديل_الملكية في الخصائص رقم الملكية ورقم مالك العقار
والايجار الشهري القديم والايجار الشهري الجديد وتاريخ التعديل، ويشغل القادح بعد
AFTER القيام المستخدم بعملية التعديل UPDATE في جدول الملكية.
بمعنى بعد اتمام عملية التعديل في جدول الملكية يقوم القادح بإدخال بيانات جدول
الملكية قبل تعديلها في جدول تعديل_الملكية.

مثال 2

```
DELIMITER $$  
CREATE TRIGGER الملكية_بعد_التعديل AFTER UPDATE ON الملكية  
FOR EACH ROW  
BEGIN  
    INSERT INTO تعديل_الملكية VALUES ( OLD.رقم_الملكية, OLD.رقم_مالك_العقار ,  
    NOW(), OLD.الإيجار_الشهري, NEW.الإيجار_الشهري );  
END $$  
DELIMITER ;
```



مثال 2

الشكل التالي يبين عملية التعديل في سجلات جدول الملكية.

```
UPDATE SET الملكية = "زناته" WHERE عنوان_الملكية * 0.1 = الإيجار_الشهري = الإيجار_الشهري
```

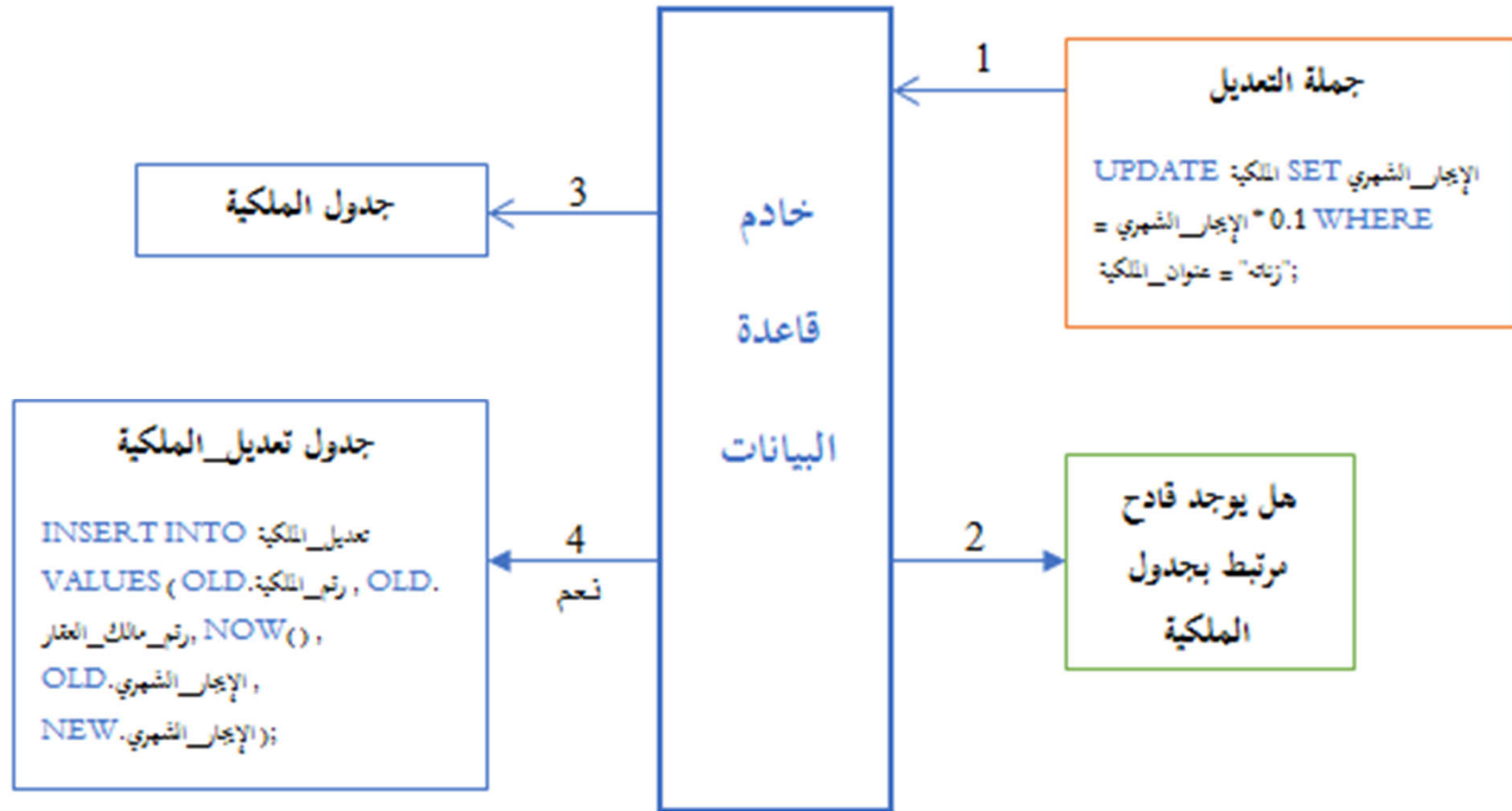
نتاج تنفيذ الأمر في الشكل السابق ، يظهر في الجدول التالي:

رقم_الملكية	عنوان_الملكية	الإيجار_الشهري	رقم_مالك_العقار
10	السراج	1400	1
20	بن عاشور	900	2
30	قرقارش	1900	2
40	زناته	1540	3
50	النوفلين	1200	1

سجلات جدول تعديل الملكية

رقم_تسلسل	رقم_الملكية	رقم_مالك_العقار	التاريخ	الإيجار_القديم	الإيجار_الجديد
1	40	3	2020-11-26	1400	1540

مثال 2 - شكل يبين خطوات تشغيل القادح بعد التعديل



عرض وحذف TRIGGER

▶ يمكن معرفة القوادح المخزنة داخل قاعدة البيانات، لعرض القوادح في قاعدة البيانات
نستخدم الأمر في الشكل:

```
SHOW TRIGGERS ;
```

▶ لعرض تركيبة قوادح بعد التعديل في قاعدة البيانات نستخدم الأمر في الشكل

```
SHOW CREATE TRIGGERS ;
```

▶ لحذف القوادح من قاعدة البيانات نستخدم الأمر في الشكل

```
DROP TRIGGERS ;
```

مزايا القادح TRIGGER Advantages

تخزين القوادح داخل قاعدة البيانات يوفر المزايا التالية:

1. يخزن القادح (المحفز) مرة واحدة مع الجداول في قاعدة البيانات، بالتالي يقلل الحاجة إلى التعليمات البرمجية الزائدة في التطبيقات.
2. تخزين القادح داخل قاعدة البيانات يتيح DBMS في منح صلاحيات الوصول الآمن إليه .
3. القادح مفيد مع الشبكات للتقليل من تبادل البيانات بين الشبكات، يمكن تنفيذ العديد من العمليات داخل الخادم بطلب واحد من العميل .
4. تواجد القادح في قاعدة البيانات يتيح السهولة في تعديله في مكان واحد فقط ويكون بعد التعديل متاح للجميع التطبيقات .

عيوب القادح TRIGGER Disadvantages

تركيبة القوادح تترتب عليها عدة عيوب وهي:

1. تشغيل القادح قد يتسبب في تشغيل قادح آخر وهكذا، بالتالي من الصعب توقع التأثير الذي قد يحدث على الجداول في قاعدة البيانات .
2. لا يمكن التحكم في تشغيل القادح لأنها تنفيذ في حالة وقوع حدث معين على الجدول المرتبطة به .
3. تعدد أشكال تركيبة القادح من نظام إدارة قواعد البيانات DBMS الى آخر قد يسبب في صعوبة تنفيذه على نظام DBMS آخر مختلف .



ملخص Summary

- ▶ أحيانا نحتاج لإجراء بعض العمليات على الجداول بشكل آلي عند إجراء عملية Insert, Update أو Delete على جدول معين، يتم ذلك باستخدام القوادح Triggers.
- ▶ تختلف بنية القادح من نظام DBMS إلى آخر، ينصح دائما بالاطلاع على التوثيق الخاص بنظام DBMS.
- ▶ يوجد معاملان NEW و OLD يستخدمان مع القادح للوصول إلى قيم الخصائص قبل وبعد التحديث في الجدول.
- ▶ كما للقادح عيوب كذلك له مزايا منها التقليل من تبادل البيانات بين الشبكات، كما يقلل الحاجة إلى التعليمات البرمجية الزائدة في التطبيقات.

نهاية المحاضرة

