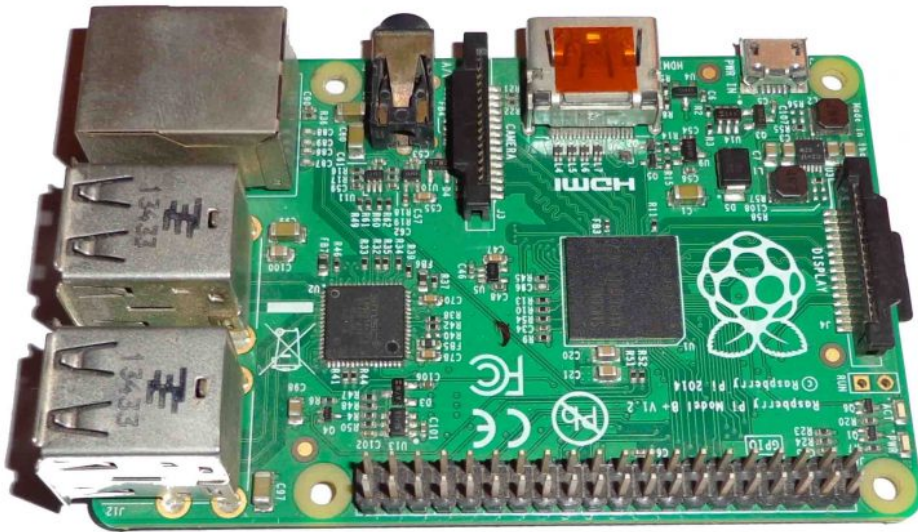
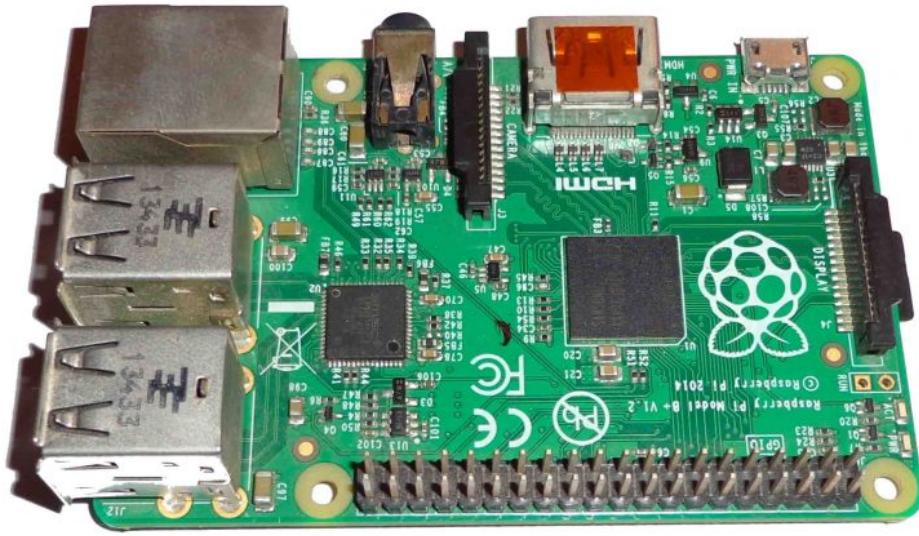


تنفيذ برنامج البايثون تلقائياً على الراسبيري باي

فى هذا الدرس سنتعلم كيفية جعل برامج البايثون تعمل بشكل تلقائي أو فى وقت محدد على الراسبيري باي باستخدام طريقة Crontab وهذه تعتبر من أهم الخطوات لى تصبح الراسبيري باي منصة مستقلة لمشاريع الأنظمة المدمجة.



المكونات المطلوبة



راسبيري باي

الكود البرمجي

في البداية نقوم بفتح ال Terminal الخاص ب الراسبيري باي وكتابة الأوامر التالية أو نقوم بفتح نافذه ال SSH الخاصة بها من جهاز آخر مربوط معها على نفس الشبكة كما تم شرحه في الدرس الخامس.
نقوم بفتح محرر النصوص Nano ويقوم بإنشاء ملف البرنامج الذى نريد تشغيله مع فتح الراسبيري باي وليكن اسمه print.py

```
nano print.py
```

نقوم بكتابة الكود بداخله بلغة البايثون python وللتوضيح سأقوم بكتابة برنامج يقوم بطباعة كلمة Hello World! ويقوم بطباعة الوقت والتاريخ الحالى.

```
#!/bin/python
import datetime
print "Hello World!"
print datetime.datetime.now()
```

ثم نقوم بالضغط علي Ctrl+O ثم enter ثم Ctrl+X وذلك لحفظ الملف السابق.
نقوم بإنشاء directory جديد لنضع الكود بداخله ونسميه launcher ونوجه ال terminal لهذا المسار الجديد ونقوم بإنشاء برنامج .shell script

```
mkdir launcher
cd launcher
```

```
nano launcher.sh
```

فى هذا البرنامج نكتب الكود التالى الذى يقوم بتوجيه الـ terminal إلى مكان تواجد كود البايثون المراد تنفيذه وكذلك بدأ تشغيله.

```
#!/bin/bash
cd /
cd home/pi
sudo python print.py
```

ثم نقوم بالضغط علي Ctrl+O ثم enter ثم Ctrl+X وذلك لحفظ الملف السابق.
نحتاج الآن أن نجعل هذا البرنامج executable كالتالى.

```
chmod 755 launcher.sh
```

نقوم بإنشاء directory جديد بإسم logs فى الـ home وهو لكى يقوم الـ crontab بكتابة أى رسائل أخطاء هناك.

```
cd
mkdir logs
```

الـ crontab هو عبارة عن daemon يعمل فى خلفية نظام الـ Linux وهو يتيح لنا تنفيذ برامج فى أوقات محددة.
نقوم بكتابة الأمر التالى لكى يفتح نافذة الـ crontab

```
sudo crontab -e
```

نقوم بكتابة الأمر التالى والذى يقوم بتشغيل الـ script مع كل Reboot للراسبيري باي وسوف تلاحظ أنه أمر بسيط جداً فقط نقوم بكتابة @ ثم متى نريد تنفيذ هذا الأمر ثم مسار الملف المراد تنفيذه وهنا كي يتضح المثال سنقوم بعمل حفظ لناتج تنفيذ الأمر فى file بإسم cronlog فى directory الـ logs الذى قمنا بإنشائه مسبقاً عن طريق (>) وهى تقوم بالإستمرار فى الحفظ فى نفس الملف ثم نقوم بعمل redirection للـ standard error فى الـ standard out عن طريق كتابة 2>&1.

```
@reboot sh /home/pi/launcher/launcher.sh >>/home/pi/logs/cronlog 2>&1
```

نقوم بعمل إعادة تشغيل للراسبيري باي عن طريق الأمر التالى.

```
sudo reboot
```

إذا قمنا الآن بفتح الفايل cronlog سنجد أنه قد طبع بداخله كلمة Hello World! ومن بعدها التاريخ والوقت ، وفى كل مرة نعيد تشغيل الجهاز فيها سيقوم بنفس الكتابة مجدداً مما يعنى أن الـ crontab يعمل جيداً.
الآن إذا أردت تنفيذ أمر معين فى وقت محدد كل ما عليك هو إتباع الـ format التالى فى الأمر.

```
MIN HOUR DOM MON DOW CMD
```

الوصف	القيم المتاحة	الخانة
الدقائق	0 59 to	MIN
الساعات	0 23 to	HOUR
يوم فى الشهر	1 31 to	DOM
الشهر	1 12 to	MON

to 6 0	يوم فى الأسبوع	DOW
-	الأمر المراد تنفيذه	CMD

وإذا أرت التكرار بشكل منتظم قم بإستبدال الخانة المراد التكرار فيها بـ * بدلاً من القيمة.
فمثلاً إذا أردت تنفيذ البرنامج السابق فى وقت محدد وكتبت التالى فى الـ crontab.

```
30 08 10 06 * /home/pi/launcher/launcher.sh
```

فهذا يعنى التالى :

30 => 30 دقيقة

08 => الساعة الثامنة صباحاً

10 => اليوم العاشر من الشهر

06 => شهر يونيو

* => أى يوم من أيام الأسبوع

مثلاً أيضاً إذا أردت تنفيذ برنامج كل دقيقة فما عليك إلا الكود التالى.

```
* * * * * /home/pi/launcher/launcher.sh
```

أما إستخدام @ يكون كالتالى:

المعنى	يساوى	الأمر
كل عام	* 1 1 0 0	yearly@
كل يوم	* * * 0 0	daily@
كل ساعة	* * * * 0	hourly@
مع كل إعادة تشغيل	-	reboot@