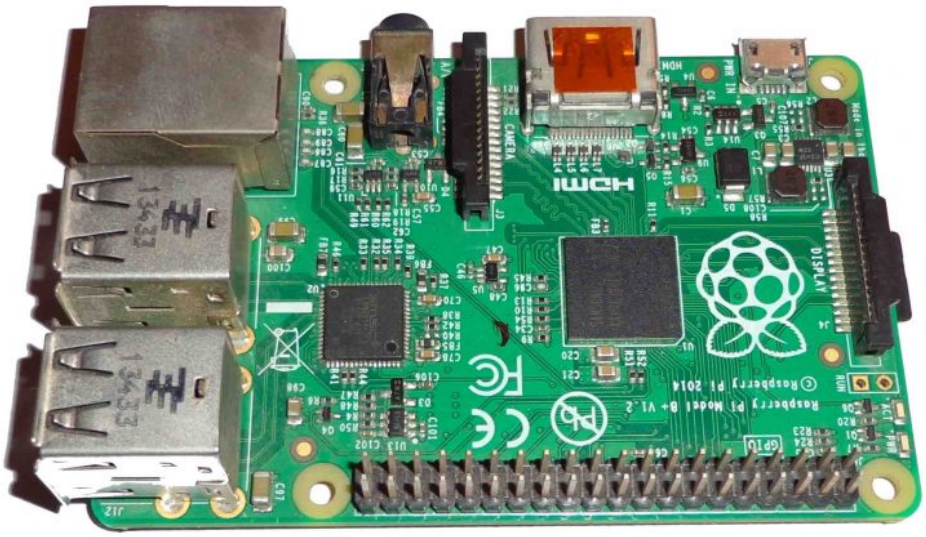
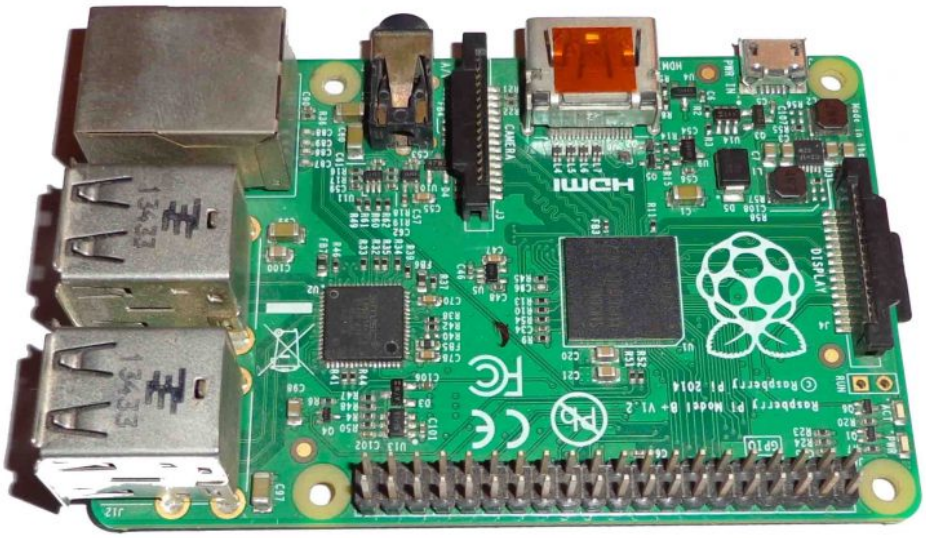
تنفيذ برنامج البايثون تلقائياً على الراسبيري باي

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



المكونات المطلوبة



راسبيري باي

الكود البرمجى

فى 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 شرحة فى الدرس الخامس.
نقوم بفتح محرر النصوص Nano ويقوم بإنشاء ملف البرنامج الذى نريد تشغيلة مع فتح الراسبري باي وليكن اسمه print.py

```
nano print.py
```

نقوم بكتابة الكود بداخلة 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 بداخلة ونسمية launcher ونوجه الـ 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 excutable كالتالي.

```
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 deamon يعمل فى خلفية نظام الـ Linux وهو يتيح لنا تنفيذ برامج فى أوقات محددة.
نقوم بكتابة الأمر التالى لكي يفتح نافذة الـ crontab

```
sudo crontab -e
```

نقوم بكتابة الأمر التالي والذى يقوم بتشغيل الـ script مع كل Reboot للراسبري 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 أعادة تشغيل للراسبري باى عن طريق الأمر التالي.

```
sudo reboot
```

إذا قمنا الآن بفتح الـفايل cronlog سنجد أنه قد طبع بداخلة 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 التالي فى الأمر.

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

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

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

وإذا أرت التكرار بشكل منتظم قم بإستبدل الخانة المراد التكرار فيها بـ * بدلاً من القيمة.
فمثلا 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 أذا أردت تنفيذ برنامج كل دقيقة فما عليك إلا الكود التالي.

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

أما إستخدام @ يكون كالتالي:

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