



01. Автоматизация с помощью скриптов на bash

№ 01.0.

1. Ознакомиться с содержимым каталога `/home/pi/CodeClub-IoT/` – там лежат учебные и справочные материалы по всем занятиям кружка.
2. Ознакомиться с содержимым каталога `/home/pi/Documents/MagPi/` – там лежат учебные материалы на английском языке из журнала **MagPi**:
 - **Essentials_*** – тематические учебники;
 - **Projects_Book_v*** – ежегодные сборники проектов (со ссылками);
 - **MagPi_*** – номера официального ежемесячного журнала о **Raspberry Pi**.

*Чтобы копировать исходные тексты программ из учебных материалов (Ctrl-C + Ctrl-V), открывайте PDF-файлы программой **Okular**, а не **qpdfview**. (Можно выбрать программу в контекстном меню, открывающемся по нажатию на правую кнопку мыши.)*

Практические задания.

№ 01.1.

1. Запустить терминальное окно, выполнить следующие действия с помощью команд.
2. Создать в каталоге `/home/pi` подкаталог **projects**
`mkdir /home/pi/projects`
3. Перейти в него (сделать его рабочим каталогом)
`cd ~/projects`
4. Создать в нём командный файл **hello.sh**
`touch hello.sh`
5. Добавить в него 2 строки (открыв в текстовом редакторе из файлового менеджера)
`#!/bin/bash`
`echo Привет, Raspberry Pi!`
6. Сделать командный файл исполняемым
`chmod a+x hello.sh`
7. Проверить его свойства (в терминальном окне) командой
`ls -l hello.sh`
(Исполняемые файлы должны иметь признак «x» в поле признаков (`-rwxr-xr-x`) и они обычно выделяются зелёным цветом в списке файлов.)
8. Запустить командный файл (скрипт) на выполнение в терминальном окне и посмотреть результат его работы
`./hello.sh`
9. Запустить скрипт на выполнение 2 раза с перенаправлением (>) вывода в файл (а не на экран), посмотреть результат:
`./hello.sh > hello.txt`
`cat hello.txt`
10. Запустить скрипт на выполнение 2 раза с дозаписью (>>) вывода в файл, посмотреть результат:
`./hello.sh >> hello.txt`
`cat hello.txt`

Подробнее о перенаправлении ввода-вывода (>, >>, <, |) можно прочитать в справочном файле по командам `/home/pi/CodeClub-IoT/theory/IoT-Shell_commands.pdf`.

1. Создайте в рабочем каталоге скрипт `temperature.sh` для измерения температуры SoC

```
#!/bin/bash
while true
do
    temp=`cat /sys/class/thermal/thermal_zone0/temp`
    echo Температура: $temp
    sleep 1
done
```
2. Сделайте его исполняемым.
3. Выполните его и посмотрите результат.
4. Прервите его бесконечное выполнение, нажав клавиши `Ctrl+C`
5. Измените скрипт, добавив условие, в котором проверяется пороговое значение (измените число по своему усмотрению)

```
echo Температура: $temp
if [ $temp -ge 58000 ]
then
    echo Перегрев процессора!!!
fi
sleep 1
```

6. Выполните его и посмотрите результат.

Подробнее об условиях и сравнениях (`-gt`, `-ge`, `-lt`, `-le`, `-ne`, `-eq`, ...) – в справочном файле `IoT-Shell_commands.pdf`.