Introduction to Rasberry Pi

Form up to 6 Teams

If using windows download Putty or enable Windows Bash

By Akash Kumar and Pallak Anand

Protocols

-SSH -- Lets you send commands remotely

-FTP -- Lets you send Files Remotely

-SFTP -- Lets you send Files Remotely over the SSH Port

-RDP -- Lets you view windows desktops remotely

-VNC -- Lets you view Unix desktops remotely

Commands “The CLI”

-sudo -- “sudo rm -rf system32” -- Superuser

-ls -- “lsusb” -- List files in a directory

-cd -- Move into or out of a directory

-mv -- move

-rm -- remove

-cp -- copy

-Nano -- Easy Text Editor

Setup (We already did this, You're welcome!)

-Download & Install the Image

-Turn on SSH

-Connect to Wifi

-Configure the OS (Config and update)

Dealing with roadblocks

NYU Doesn’t let us use Bonjour, so we must improvise!

-Running programs on startup

- Wifi vs ethernet

-Questions?

Now onto the active part.. Where y’all get to do stuff

IP address script

Navigate to the python script and change the email to your own. Also rename the pi to the number written on it, so we can tell who’s pi is who’s later.

SSH

Windows : Putty

*nix Terminal ssh user@pi

Expand the Filesystem

sudo raspi-config

Package Manager

apt-get moo

Google Assistant API

- Google Assistant Library for Python
  - google.assistant.library package
  - class google.assistant.library.Assistant(credentials)

Setup account

1. Create google developer account
    a. Step 1: Sign up for a Google Play Developer account.
    b. Step 2: Accept the Developer Distribution Agreement.
    c. Step 4: Complete your account details.
2. Open google cloud platform

Setup assistant

1. Go to API manager
2. Enable Google Assistant API
3. Go to “Credentials" and set up OAuth Content Screen
4. Make sure that you enable Web & App Activity, Device Information and Voice & Audio Activity in Activity Controls for the account.
5. Go to “Credentials” tab and Create new OAuth client ID
6. Select application type as “Other” and give the name of the key
7. Download the JSON file that stores the OAuth key information and keep it safe

Configure hardware

1. `(env) python -m pip install --upgrade google-auth-oauthlib[tool]`
2. (env) google-oauthlib-tool --client-secrets "PATH_TO_YOUR_JSON_FILE" --scope https://www.googleapis.com/auth/assistant-sdk-prototype --save --headless
3. `Please go to this URL: https://...`
4. `Enter the authorization code:`
5. Connect your LED between GPIO pin 25 and ground.
6. `pip install RPi.GPIO`
7. `cd /home/pi`
8. `sudo nano main.py`

Testing!

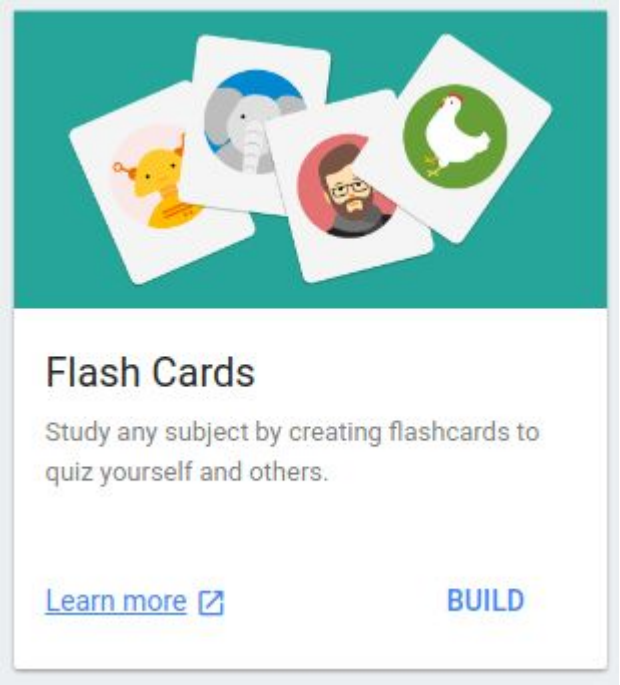
Creativity is key

You don’t even have to code!

Think of something with your team

5 minutes

Write down your q’s&a’s



Example Documentation

https://developers.google.com/actions/

-Poke around and see what project you want to try to implement

-Use the github repo for pre-existing projects

-Use the templates dropdown for easy instructions

That’s it!

… now try some stuff