Lesson 9

(5 min)
Logistics

- Check in —
  - Who doesn't have Minecraft Helper working?
- Did anyone get to extend it?

(15 min)
SpaceGeek Share out

- Each student presents his/her skill extensions
- What are the 3 new features you added?
- Why did you choose to add these?

(10 min)
API Discussion — look at student provided examples

- What is an API?
  - Application Programming Interface
  - Set of predefined functions (or routines / protocols / tools) that perform specific tedious and labor intensive tasks
  - Leverage these pre-developed functions from within your code (so that you don't have to reinvent the wheel)
  - Allows you to **delegate**
  - AND what **you** can do is make the data easier to work with for your users

- How do you make this API calls?
  - HTTP — Hypertext Transfer Protocol
  - We use this protocol to transfer information from one service to another (in this case, the API service and our Alexa skill)
  - Two HTTP Request Methods (among others)— think of these as verbs
    - **GET** — requests data from a specified resource
    - **POST** — submits data to be processed to a specific resource
    - https://www.w3schools.com/TAgs/ref_httpmethods.asp

- What makes for a usable API (for our current use, based on our set of skills)?
  - Has to communicate over HTTP — specifically GET verb
  - Preferably returns data in JSON
  - **Well documented** — is clear on what to do & has examples in Java

- Examples
  - Good — YouTube Data API
  - Not as good — Wikipedia

(15 min)
Zoom – breakout groups

- Work with partner to pick an API that you like
- How would you leverage this API from within an Alexa Skill?
- What would your interaction with the user look like?
- Document in Trello using draw.io

(10 min)
Wrap Up — Class Share

- What APIs did you pick?
- What would your skill do?
- How would the user interact with it?
- Peer Feedback?