PIC 16, Winter 2018 – Preparation 7W

Assigned 2/14/2018. To be completed by class 2/21/2018.

Intended Learning Outcomes

By the end of this preparatory assignment, students should be able to:

- create an intent schema,
- define custom slot types,
- write sample utterances,
- write code to interpret a request from Alexa, and
- write code to return a valid response to Alexa.

Tasks

- ☐ Read all of Defining the Interaction Model in JSON and Text. If you feel lost, it's likely that some of the concepts from last time are missing; you might want to review that material. As you're reading, you don't need to click all the links, but you might want to skim:
  - The Built-in Intent Library, and especially the Standard Built-in Intents, then Implementing the Built-in Intents. You don't need to memorize these lists, but you should definitely know that there are built-in intents for "help", "cancel", "yes", "no", and "stop". Also, you should know to check these references for a built-in intent before creating a duplicate. It's not required to use built-in intents when they exist, but using built-in intents can save you time and make intent recognition more reliable.
  - Slot Type Reference (especially Numbers, Dates, and Times). You probably *should* memorize the names of built-in slot types for "date", "time", "number", and "duration". Also, you should know to check these references for a built-in slot types before creating your own custom slot type. As with intents, it's not required to use built-in slot types when they exist, but it can save you time and make slot value recognition more reliable.
  - Best Practices for Sample Utterances and Custom Slot Type Values, especially if you're looking for recommendations about what different variations you should include to make intent recognition reliable.
- ☐ After reading all this, here's a situation to consider.
  You're creating a skill called "Animal of the Day". Each day, a different animal is featured, but your skill can also tell the user about specific animals. It should have responses to the following:
  - Alexa, open Animal of the Day
  - Alexa, ask Animal of the Day what is today's animal?
  - Alexa, ask Animal of the Day about the animal of the day.
  - Alexa, ask Animal of the Day what the animal was on October 31, 2017.
  - Alexa, ask Animal of the Day about polar bears.
  - Alexa, ask Animal of the Day when elephants have been featured.

  For practice, write an intent schema, define any custom slots (if necessary), and compose a few sample utterances for each intent (encompassing the phrases above and more). This is an open-ended problem that you don't have to turn in. I won't post a "solution" but I'd be happy to review your attempt in office hours.
- ☐ You still don't necessarily know what to *do* with an interaction model in order to make a skill. That's ok! Being able to write it is the hard part; we'll got through where all this information goes in Tutorial 6M.
- ☐ Defining the interaction model is half the battle! Once your interaction model can be used to map an utterance to an intent (and slot values), all that's left is for your cloud-based service to convert

Alexa's JSON request to a JSON response. You already know how to navigate Python dictionaries and lists, so all you need to learn is the standard request structure and the standard response structure. Take a look at Handle Requests Sent by Alexa.

Of course you're not interested in the *syntax* for working with Java, but it's still good to take a look at the logic. For instance, you can't use Java's `onLaunch` method, but you will still need to know whether a request is a `LaunchRequest`. For that, you'll just need to look through the JSON directly, as described in Determining the Request Type (Other Languages).

Assume you accept the request into your cloud-based service function as a parameter called `request`. Fortunately, the JSON will have already been converted to a dictionary for you, so to check if it's a `LaunchRequest`, the Python code will be:

```
if request["session"]["request"]["type"] == "LaunchRequest":
```

Although this document doesn't tell you much about the rest of the request or response formats, it gives you an idea for the sort of things your skill will need to do.

- Finally, what you've been waiting for: the Request and Response JSON Format. Just skim it. We'll go over the important parts in lecture. But based on the reference, think about whether you can write code to:
    - determine whether the session is new or not;
    - check the application ID the request is coming from;
    - determine whether the request type is `IntentRequest`, and if so, get the name of the intent;
    - if the intent name is "`MyIntent`", get the value of slot "`MySlot`" if it's present; and
    - create a response object:
        - that will end the session, and
        - cause Alexa to say "Goodbye!"
- If you feel ready and want to get started toward the assignment, you're welcome to start the step by step tutorial for Deploying a Skill. The assignment will be to customize the skill as described at the end; specific instructions and criteria for self-assessment will be released during class.