

## Article

# Intent Phrases

The keys that you include in your global vocabulary file to show how users engage your app from Siri.

## Overview

The `IntentPhrases` key contains sample phrases that a user might speak to initiate the handling of one of your intents. The value of this key is an array of dictionaries, each of which defines one of the sample phrases to speak. Each dictionary contains the following keys:

### `IntentName`

(Required) A string containing the name of the intent class to which the examples apply. For example, you might specify the string `"INRequestRideIntent"` to specify sample phrases for booking a ride.

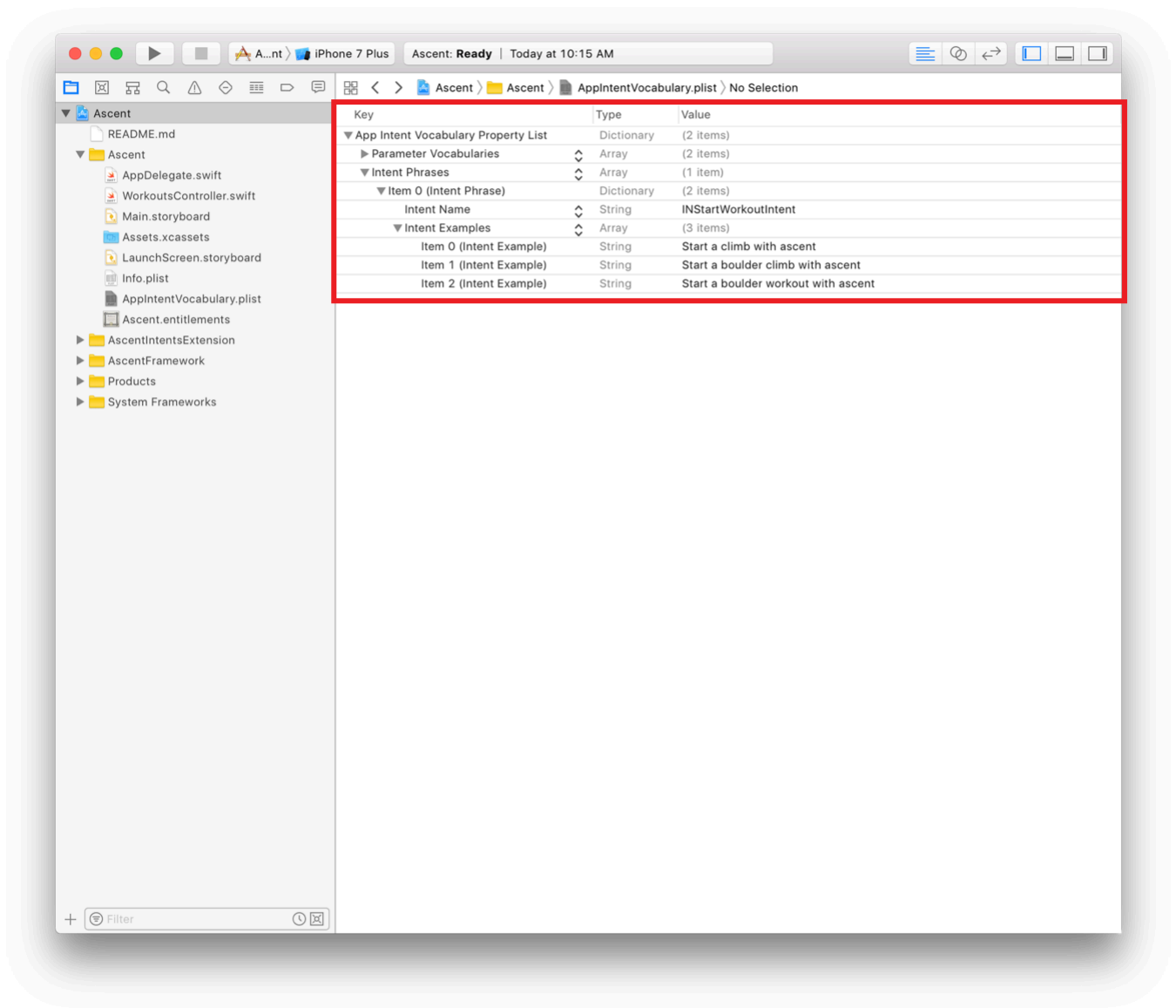
### `IntentExamples`

(Required) An array of strings containing the sample phrases that apply to the specified intent. Your sample phrases may be as complex as needed, but should reflect actual phrases that users might say. You may include custom terminology that you defined in the [Parameter Vocabularies](#) portion of your vocabulary file.

### Important


Apps are expected to provide some example phrases. Siri displays your app's sample phrases in Siri Guide so that users can access them directly from the Siri interface.

Xcode helps you add example phrases to your vocabulary file. When you create a property list file with the name `AppIntentVocabulary.plist`, Xcode knows that the file should contain custom vocabulary, and it makes only the relevant keys available in the property list editor. The figure below shows an example of a global vocabulary file in Xcode that contains sample phrases for starting a workout.





# See Also

## Related Documentation

-  [Parameter Vocabularies](#)  
The keys you include in your global vocabulary file to describe app-specific terms.

## Articles

-  [Adding User Interactivity with Siri Shortcuts and the Shortcuts App](#)  
Add custom intents and parameters to help users interact more quickly and effectively with Siri and the Shortcuts app.
-  [Defining Relevant Shortcuts for the Siri Watch Face](#)  
Inform Siri when your app's shortcuts may be useful to the user.



## Deleting Donated Shortcuts

Remove your donations from Siri.



## Dispatching intents to handlers

Provide SiriKit with an intent handler capable of handling a specific intent.



## Improving Siri Media Interactions and App Selection

Fine-tune voice controls and improve Siri Suggestions by sharing app capabilities, customized names, and listening habits with the system.



## Improving interactions between Siri and your messaging app

Donate app-specific content, use Siri's contact suggestions, and adopt the latest platform features to create a more consistent messaging experience.



## Registering Custom Vocabulary with SiriKit

Register your app's custom terminology, and provide sample phrases for how to use your app with Siri.



## Confirming the Details of an Intent

Perform final validation of the intent parameters and verify that your services are ready to fulfill the intent.



## Handling an Intent

Fulfill the intent and provide feedback to SiriKit about what you did.



## Resolving the Parameters of an Intent

Validate the parameters of an intent and make sure that you have the information you need to continue.



## Generating a List of Ride Options

Generate ride options for Maps to display to the user.



## Handling the Ride-Booking Intents

Support the different intent-handling sequences for booking rides with Shortcuts or Maps.



## Donating Reservations

Inform Siri of reservations made from your app.



## Specifying Synonyms for Your App Name

Provide alternative names for your app that are more familiar or easier for users to speak.



## Localizing Your Vocabulary for Chinese Dialects

Apply emphasis markers to your pronunciation tips to assist Siri with Chinese dialects.