Headlines App Key Features

- ☑ Glance
    - ☑ Utility
        - ☑ Show up to 4 feeds on the glance
        - ☑ Tap on a headline to open the app and prepare for handoff to your phone.
    - ☑ Design
        - ☑ Upon glance load, iterate through each headline, grab the latest title for the feed and it to an array of results to be passed to the watch and updated all at once.
- ☑ App on Watch OS
    - ☑ Views
        - ☑ Secondary (upon tapping a headline or being passed from the glance)
            - ☑ Utility
                - ☑ Display the title of the headline and show the lead of the article and indicate that you can use handoff to view the rest.
            - ☑ Design
                - ☑ Receive an Article object from the glance, display the headline and the first line of the article.
        - ☑ Main
            - ☑ Utility
                - ☑ Inherits utility from the glance
            - ☑ Design
                - ☑ Inherits utility from the glance
                - ☑ Consider revising design after real word usage
- ☑ App on iOS
    - ☑ View
        - ☑ Main
            - ☑ Utility
                - ☑ A navigation controller with a table that displays the subscribed feeds and their friendlyName's.
- ☑ Models
    - ☑ Feed
        - ☑ Properties
            - ☑ friendlyName
                - ☑ This property is retrieved from the title tag of the feed
            - ☑ url
                - ☑ The feed URL
            - ☑ insertManagedFeedInContext:
                - ☑ A bridge to the object ManagedFeed. It inserts the new managed object into the current context.
    - ☑ Article
        - ☑ Protocols
            - ☑ NSCoding
            - ☑ github.com—AutoCoding
        - ☑ Properties
            - ☑ URL
                - ☑ The URL of the article.
            - ☑ headline
                - ☑ The title.
            - ☑ lead
                - ☑ This is the first two or three lines of the first paragraph from the article.
            - ☑ date
                - ☑ The time the article was posted.
- ☐ View Controllers
    - ☐ Raw Web View Controller
        - ☐ Properties
            - ☐ WKWebView articleView
            - ☐ URL
    - ☐ Read View Controller
        - ☑ Properties
            - ☑ article
                - ☑ set while preparing to segue from the main view controller
                - ☑ www.appcoda.com—storyboards-ios-tutorial-pass-data-between-view-controller-with-segue
        - ☑ Interface Items
            - ☑ WKWebView articleView
        - ☑ Methods
            - ☑ didSegue or something like init
                - ☑ set title of UINavBar to article's headline property
                    call readability

Methods

didSegue or something like init

<u>Headlines App Key Features</u>

- ☑ call readability
- ⊟ Actions
  - ☑ readability
    - ☑ use github.com—readability-objc
  - ☐ breakOut
    - ☐ segue to the full article in Raw Web View Controller
- ☐ Share Extension View Controller
  - ☐ Protocols
    - ☐ UIPickerViewDelegate
    - ☐ UIPickerViewDataSource
  - ☐ Properties
    - ☐ feeds
  - ☐ Interface Items
    - ☐ feedPicker
  - ☐ Actions
    - ☐ saveFeed
      - ☐ get the selectedRowInComponent of feedPicker and access feeds object at index of the row, with this and send it to FeedSource's addWithFeed
  - ☐ Methods
    - ☐ pickerView titleForRow forComponent
      - ☐ return the friendlyName from the feeds object at index of the row
    - ☐ numberOfRowsInComponent
      - ☐ Return feed's object count
    - ☐ loadView
      - ☐ Implementation
        - ☐ Get the input item from the extension context and pass the URL to FeedSource's findFeedsForURL. Save the results to feeds.
          - ☐ developer.apple.com—ExtensionCreation.html
  - ☐ Design
    - ☐ Present a drop down option with the URL below
- ☑ Feed Picker View Controller
  - ☑ Interface Items
    - ☑ self.table
      - ☑ This contains a list of the URLs of RSS options with friendlyNames and URLs
  - ☑ Properties
    - ☑ URL
      - ☑ from segue, this is the URL to work with
    - ☑ feeds
  - ☑ Actions
    - ☑ init
      - ☑ Call FeedSource's findFeedsForURL and store the results in feeds
    - ☑ subscribe
      - ☑ Call FeedSource's addWithURL
- ☑ Feed Table View Controller
  - ☑ Properties
    - ☑ feedSource
  - ☑ Actions
    - ☑ addFeed
      - ☑ Implementation
        - ☑ If the URL given isn't an RSS feed and there's more than one, segue to feed picker view controller and show the feed options.
        - ☑ Otherwise send the URL to the FeedSource's addWithURL
  - ☑ Methods
    - ☑ prepareForSegue
      - ☑ send URL to feedPickerViewController
    - ☑ readArticleFromHandoff
      - ☑ this is called from the app delegate and has an Article object, segue to the Read View Controller and open the article
    - ☑ Also Apple fetchedResultsController delegate boilerplate! This is for updating. www.raywenderlich.com—core-data-tutorial-for-ios-how-to-use-nsfetchedresultscontroller
    - ☑ tableView tableView numberOfRowsInSection
      - ☑ Implementation
        - ☑ Return the number of objects from the section using the number of Objects from fetchedResultsController
    - ☑ configureCell atIndexPath
        Implementation

Headlines App Key Features rowAtIndexPath
- ☑ Implementation
  - ☑ Retrieve corresponding Feed object from fetchedResultsController and display the friendlyName as title.
- ☑ Handoff Controller
  - ☑ Interface Items
    - ☑ headline
    - ☑ lead
  - ☑ Properties
    - ☑ article
- ☐ Glance
  - ☑ Properties
    - ☑ ticker
  - ☑ Interface Items
    - ☑ headlineTable
  - ☐ Methods
    - ☐ refresh
      - ☐ Only refresh the tables if there are new headlines.
      - ☑ Call ticker's refreshHeadlines object and use the completion block and assign the headline property of the Article object to the titles in the headlineTable. If nothing returns default to an item with "No headlines"
    - ☑ table didSelectRowAtIndex
      - ☑ developer.apple.com—TheGlanceController.html
      - ☑ get the Article object at the current index from articles and call updateUserActivity userInfo webPageURL with the URL of the Article and the Article object as the userInfo
    - ☑ willActivate
      - ☑ Usage
        - ☑ Update the glance
      - ☑ Implementation
        - ☑ call refresh
    - ☑ awakeWithContext
      - ☑ Usage
        - ☑ This is called when the glance is woken up for the first time
      - ☑ Implementation
        - ☑ If there are no objects in the results, default back one item with "No headlines"
        - ☑ Create a Ticker object and store as ticker. Call refresh.
- ☑ Objects
  - ☑ FeedSource
    - ☑ Methods
      - ☑ +findFeedsForURL
        - ☑ find the rel = xml/rss thing for the webpage and grab the URL
        - ☑ parse the XML of the URL, grab the title tag and assign it to the friendlyName of a newly created Feed object
        - ☑ add this object to an array of feeds
      - ☑ +addWithFeed
        - ☑ add the feed object to the managedObjectContext
        - ☑ tell the counterpart watch app that we added a feed
          - ☑ message data will be a dictionary containing the operation type, like create or delete and the feed information
      - ☑ +addWithURL
        - ☑ grab the friendlyName from the URL with friendlyNameForURL, create a Feed object
        - ☑ call addWithFeed
      - ☑ removeManagedFeed
        - ☑ This is called from the table when swiping to delete, it removes this from the current context and commits the transaction to watchOS
      - ☑ friendlyNameForURL
        - ☑ get the friendlyName for the feed
      - ☑ isValidURL handler
        - ☑ returns yes or no depending on whether the feed can be successfully parsed
    - ☑ Properties
      - ☑ table
      - ☑ managedObjectContext
      - ☑ fetchedResultsController
  - ☑ Ticker
    - ☑ Methods
      - ☑ reloadTable
      - ☑ refresh
        - ☑ Usage

Methods

refresh

Headlines App Key Features

- ☑ This will clear the headlines property and replace it with a new array containing new Article objects.
  - ☑ Implementation
    - ☑ Fetch the Feed object results from the FeedSource object.
    - ☑ For each feed subscribed, download the feed, parse out the newest title into headline, the URL and the lead into a new Article object and insert into an array.
- ☑ Properties
  - ☑ headlineTable
    - ☑ Use this so we can update when we receive a WCSessionFile from the host app. We pretty much take care of updating the table in the Ticker this way.
  - ☑ feedSource