

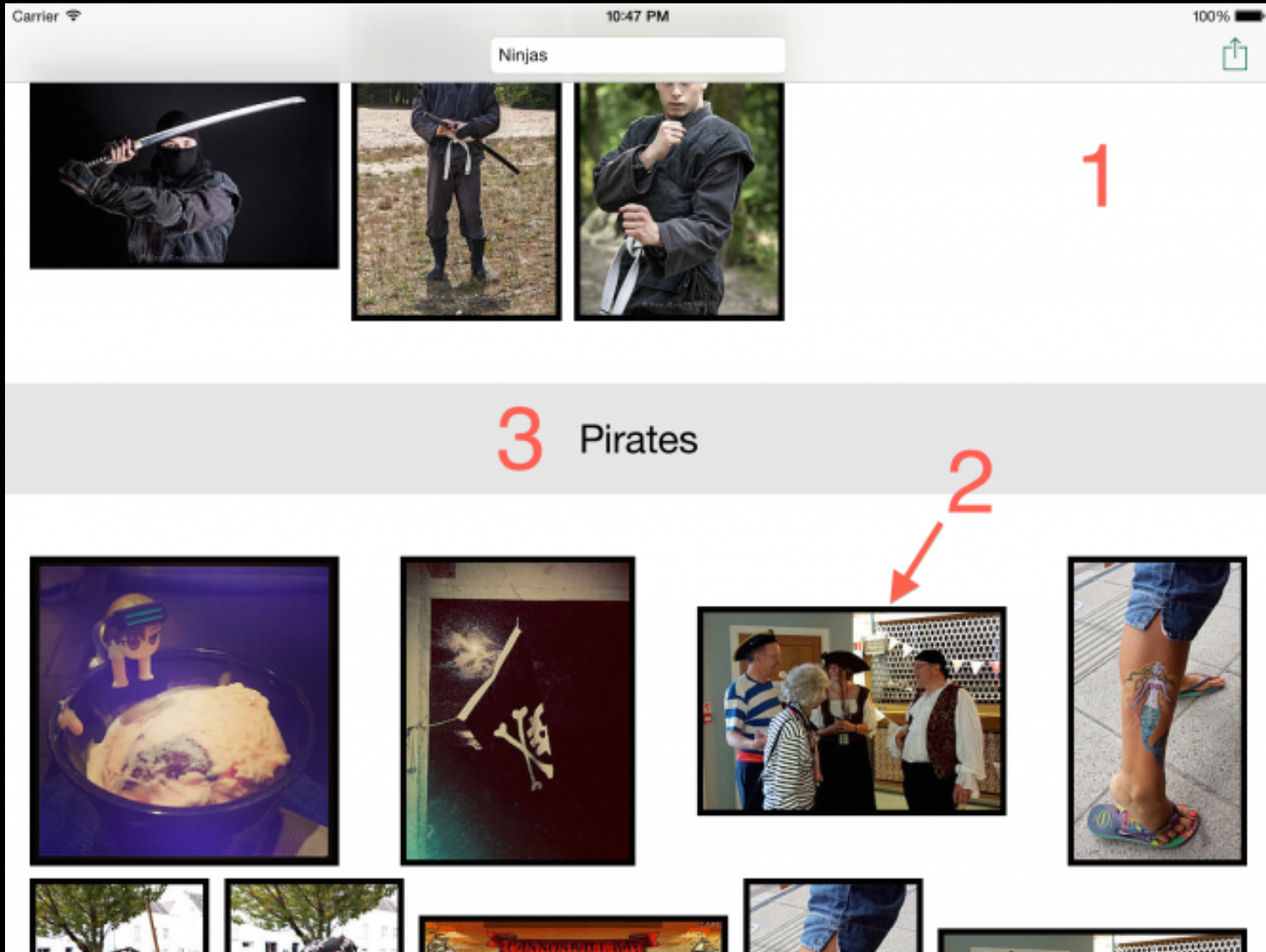
Mobile Application Development
Aileen Pierce

COLLECTION VIEWS

Collection Views

- Collection views makes it easy to display data in rows and columns to form grids
- Similar approach as the table view data source and delegate pattern
- Collection views are very customizable
- You can create your own custom layout and transitions
- Can scroll horizontally or vertically

Collection Views



Collection Views

- The collection view is the main view that the content is displayed in (1)
 - `UICollectionView` class
 - `UICollectionViewDataSource` protocol
- Cells hold the content in the view (2)
 - `UICollectionViewCell` class
- Supplementary views hold extra information that you want but not in the cells (3)
 - Headers or footers
 - `UICollectionReusableView` class

Collection Views

- Sections, items, and cells are handled similar to table views
- `collectionView(_,
viewForSupplementaryElementOfKind,
atIndexPath)`
 - Displays supplementary views
 - Use their own reuse identifier
 - Header
 - `UICollectionViewElementKindSectionHeader`
 - Footer
 - `UICollectionViewElementKindSectionFooter`

Layout

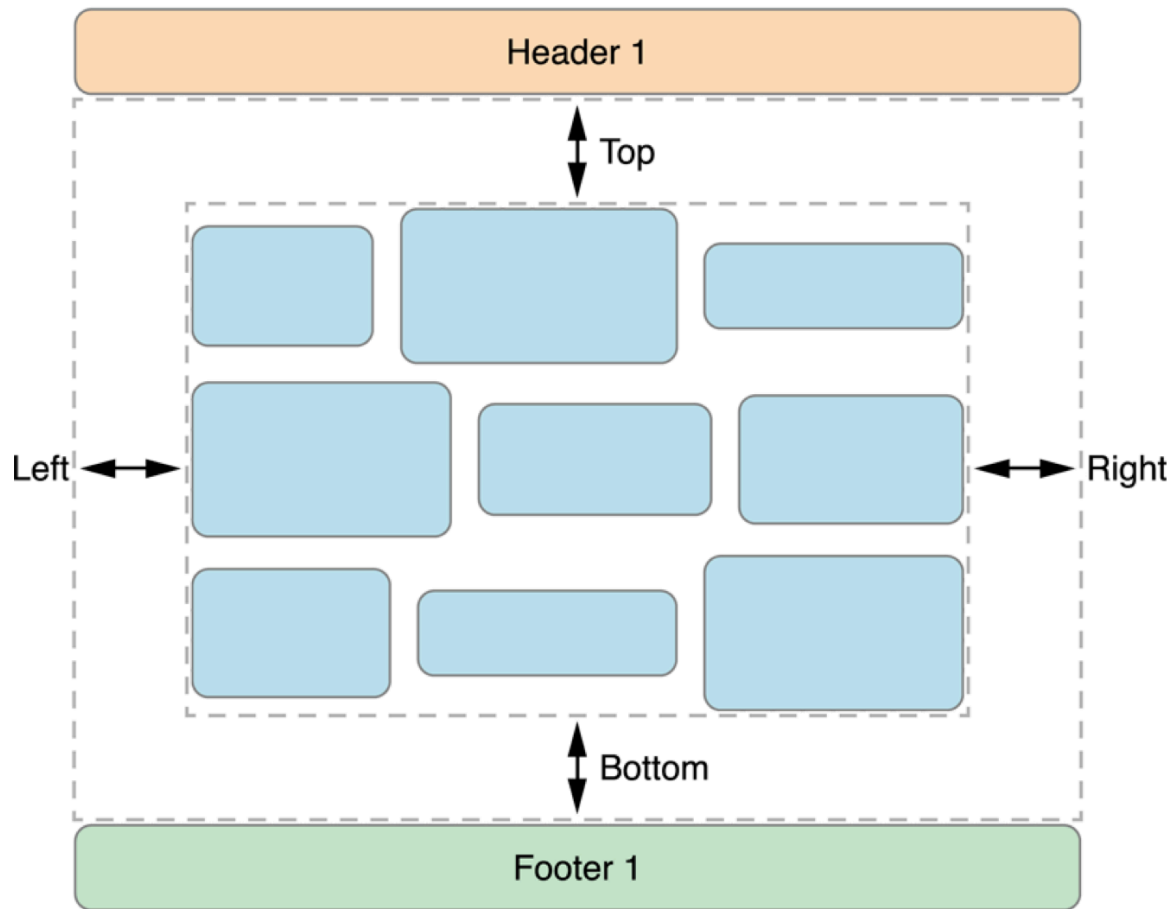
- The `UICollectionViewLayout` class lets you create custom layouts
- The `UICollectionViewFlowLayout` class provides a basic flow-based layout
 - `scrollDirection` property determines direction the view scrolls
- The `UICollectionViewDelegateFlowLayout` protocol methods let you define the size and spacing of items

Layout

- `UICollectionViewDelegateFlowLayout` protocol methods let you define the size and spacing of items
 - `collectionView(_: layout, sizeForItemAtIndexPath)` is responsible for telling the layout the size of a given cell
 - `collectionView(_: layout, insetForSectionAtIndex)` returns the spacing between the cells, headers, and footers

Layout

```
inset = UIEdgeInsetsMake(top, left, bottom, right)
```



Collection Views

