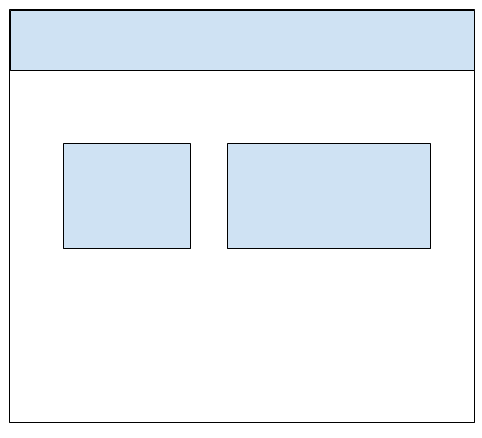
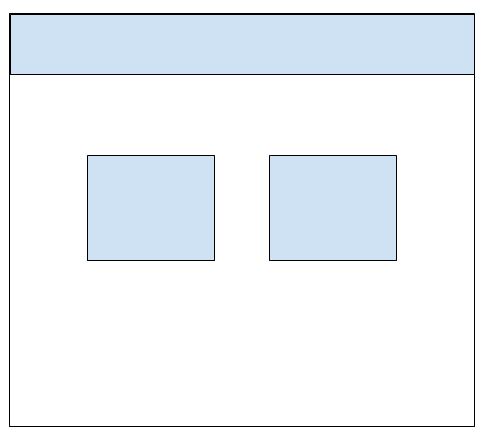
# Instructions – Exercise 5.2 – Navigation and Layout

**Layout(s)**

about



contact



in-n-out-books, part 1

**Instructions**

* Generate a new application in your week-5 directory and name it in-n-out-books
  + ng new in-n-out-booksp1
  + Accept “y” for routing
* Install Angular material
  + ng add @angular/material
  + Choose: “Custom” for the theme
  + Choose “Y” for Angular Material Typography styles
  + Choose “Y” for browser animations
* Install Angular’s flex-layout
  + npm install @angular/flex-layout
* Rename the styles.css file to styles.scss (located under the src directory
* Rename the custom-theme.scss file to \_theme.scss
* styles.scss
  + Add an import statement for the \_theme.scss file
    - @import ‘./theme’;
* angular.json
  + Update the styles references to reflect the correct file names
    - src/\_theme.scss
    - src/styles.scss
* index.html
  + Add a class reference to mat-app-background in the HTML body element
    - <body class=”mat-typography mat-app-background”>
* \_theme.scss
  + Rename each of the theme variables to $dark-<theme-name>
  + Assign $mat-blue to the primary color
    - $dark-primary: mat-palette($mat-blue)
  + Assign $mat-green, 400 to the accent color
    - $dark-accent: mat-palette($mat-green, 400)
  + Assign $mat-red, 400 to the warn color
    - $dark-warn: mat-palette($mat-red, 400)
  + Replace the light theme with the mat-dark-theme
    - $dark-theme: mat-dark-theme($dark-primary, $dark-accent, $dark-warn)
* Run and test the application to verify there are no errors
* Generate 3 new components: about, contact, and book-list
  + ng g c about
  + ng g c contact
  + ng g c book-list
* app.routing.module.ts
  + Add each components path to the routes array
    - path: ‘’, component: BookListComponent
    - path: ‘/book-list’: component: BookListComponent
    - path: ‘/contact’: component: ContactComponent
    - path: ‘/about’: component: AboutComponent
* app.module.ts
  + Add import statements for MatToolbarModule, MatCardModule, MatButtonModule, MatListModule, FlexLayoutModule, MatIconModule
  + Add the imported modules to the imports array
* app.component.html
  + Add a <mat-toolbar> with one <mat-toolbar-row>
  + Inside the <mat-toolbar-row> and a div for the logo and a div for the applications links.
  + Logo div
    - Set the height to 80
  + Links div
    - Using flex-layout, create a new row and set the alignment to “end start”
      * <div fxLayout=”row wrap” fxLayoutAlign=”end start”>
    - Add a mat-button with a routerLink for /book-list, /contact, and /about
      * <button mat-button routerLink=”/book-list”>Book List
  + Add a line break to separate the toolbar and main container
    - <br />
  + Using flex-layout create a new row and set the row to wrap
    - <div fxLayout=”row wrap”
    - Create another div and add the fxFlex directive
      * <div fxFlex
      * Note: this tells Angular flex-layout that this is a flex element under a flex-row
    - Add a <mat-card> and give it a class of main-container
    - Add a <mat-card-title> and add a data binding call to a variable named assignment
      * {{ assignment }}
    - Underneath the <mat-card-title> Add two new line breaks
      * <br /><br />
    - Add a <mat-card-content> and put the <router-outlet> inside it
* app.component.ts
  + Create a new variable and name it assignment
  + In the components constructor assign the assignment variable the value “Exercise 5.2 - Navigation and Layout”
    - Hint: you will need to create a constructor
      * constructor() {}
* app.component.css
  + .logo
    - Set the top margin to 5px
  + .main-container
    - Set the top padding to 25px
    - Width to 80%
    - Center the div
      * margin: 0 auto;
* Run and test the application
  + Verify there is a top navigation bar with a logo (top left-side) and 3 menu items (top-right)
  + Verify each link works and you can see the HTML text in the body of the main-container div
  + Verify you can see the assignment number on each page
  + Warning: Do not proceed to the next steps until the above tests pass
* about.component.html
  + Note: Using the enterprise-composer-app as a guide we are going to recreate the about and contact us pages, but this time using our favorite book instead of a composer and Angular Material components
  + Using flex-layout create a new row div and set the row to wrap
  + Inside the first row div add two more fxFlex divs (we are going to display the content in two columns: image and description)
  + Research books and select your favorite author to write a 75-word bio about the book.  Add this to the about.component.html page.  You will also need to include an image of the author, which will be placed to the left of the 75-word write-up.  Make sure the image is royalty free and include the source as a caption.
* contact.component.html
  + Update the HTML in the contact.component.html to tell users how they may contact the selected author
    - At minimum there should be two side-by-side divs.  The left <div> should contain a header icon and a street address.  The right <div> should contain a header icon and a phone number.  Use <mat-card-header> and <mat-card-content> for each div and paragraph tags for the content
* Add a new file to the app directory named book.interface.ts and name the interface IBook
* book.interface.ts
  + isbn: string
  + title: string;
  + authors: Array<string>
  + description: string
  + numOfPages: number
* Run and test the application
  + Verify the content we added to the contact page is viewable and formatted correctly
  + Verify the content we added to the about page is viewable and formatted correctly