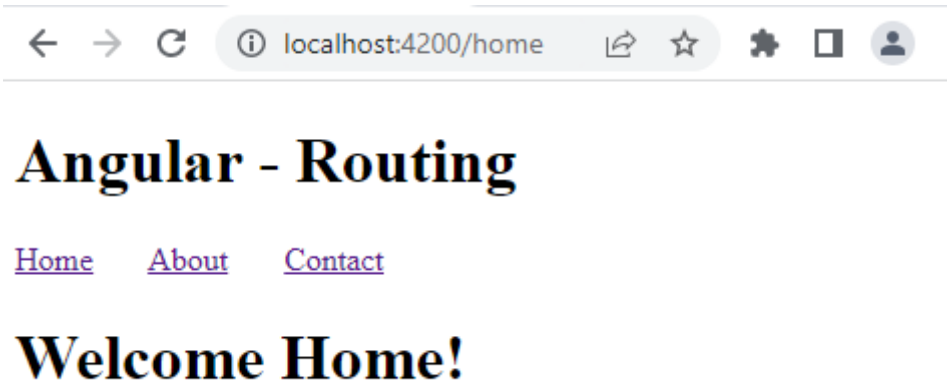
HOL-#06
=======

Q1. Your challenge is to design a routing Angular application as shown below:



[a] The home page should display the 'Angular' logo with the
following text:

The Modern Web Developer's Platform -> in <h1> element and an unordered list with the following list items:
- Developed Across All Platforms
- Speed & Performace
- Loved by Millions

[b] The about page should display -
About Angular -> in <h1> element

Angular is a development platform, built on TypeScript. As a platform, Angular includes:

With the following text in the paragraphs:

A component-based framework for building scalable web applications

A collection of well-integrated libraries that cover a wide variety of features, including routing, forms management, client-server communication

[c] The contact page should display
Your organization logo and the contact address

[d] Run your application and check it functionality.

[e] Lastly, take care of the catch-all / wildcard route.

This should render a Page 404 Error message.

Q2. Your next challenge is to implement routing in the `eComm` application.

NOTE: If you have forgotten to including Routing initially we can do it manually.

Refer to the following URL

https://www.samjulien.com/add-routing-existing-angular-project

Q3. Generate a new component for product details.

Q4. In the app module add a route for product details, with the path of products/:productid and the product details component for component

```
  imports: [
    BrowserModule,
    RouterModule.forRoot([
      { path: '', component: ProductListComponent },
      { path: 'products/:productId', component: ProductDetailsComponent},
    ])
  ],
```

- Next, open the product list component HTML file and modify the product name anchor to include `routeLink` with the `productid` as the parameter.

- Verify that the link is work.

On click the product name - It should display:

product-details-work!

- Observe the URL in the address bar.
  Record your observation.

Click the web-browser’s back button to get the product list once again.

Q5. Your next challenge!

Display the product details.

- In the product details component import the products and it’s interface.

- Define the product property in the class

- Inject `ActivatedRoute` into the constructor() by adding `private route: ActivatedRoute` as an argument

NOTE: By inject `ActivatedRoute` we are configuring the component to use the service.

- In the ngOnInit() method, extract the productid from the route parameters and find the corresponding product in the products array.

The code is show below for your reference:

```
onInit() {

  // First get the product id from the route.

  const routeParams = this.route.snapshot.paramMap;

  const productIdFromRoute =
Number(routeParams.get('productId'));



  // Find the product that correspond with the id
provided in route.
```

```
  this.product = products.find(product => product.id
=== productIdFromRoute);

}
```

- Modify the product details HTML file so that it displays the product details.
  i.e. Name, price and description.

NOTE: Use the `product` property for this purpose.

Your web-browser should now be able to display the product details when the user click the product name.

Check!