Rangle.io Interview Screening Questions

Please answer **two** questions that you feel are most appropriate for your experience.

1. Suppose `getData` is a function that takes a query object and returns a promise for the result of the query. Suppose also that `someArrayOfQueries` is an array of query objects. Explain what would be printed by the following code and why:

```
function runMultipleQueries(queries) {
  var results  = [];
  queries.forEach(doQuery);

  return results;

  function doQuery(query) {
    getData(query)
      .then(results.push.bind(results));
  }
}

function log(value) {
  console.log(value);
}

runMultipleQueries(someArrayOfQueries).forEach(log);
```

2. Refactor the following AngularJS controller.

```
angular.module('product', [])
  .controller('ProductCtrl', function ProductCtrl($http, $q) {
    var vm = this;

    function loadProduct(productId) {
      var finalProduct = $q.defer();
      $http.get('/api/product/' + productId)
        .success(function (productData) {
          $http.get('/api/productReviews/' + productId)
            .success(function (productReviews) {
              var product = {
                product: productData,
                reviews: productReviews
              };
              finalProduct.resolve(product);
            });
        });
      return finalProduct.promise;
    }

    loadProduct(1)
      .then(function (product) {
        vm.product = product;
      });
  });
```

3. Improve this React component.

```
var JSON_URL = "https://api.example.io/comments.json";

class CommentList extends React.Component {
  constructor() {
    super();
    this.state = { comments: [] }
  }

  componentDidMount() {
    $.ajax({
      url: JSON_URL,
      dataType: 'json',
      success: function(data) {
        this.setState({comments: data.comments});
      }.bind(this)
    });
  };

  render() {
    return <ul> {this.state.comments.map((comment) => {
                  return <li>{comment.body}—{comment.author}</li>;
                })}
           </ul>;
  }
}

React.render(<CommentList />, document.getElementById('root'));
```

4. Implement a function `pipe()` that takes several functions as arguments and returns a new function that will pass its argument to the first function, then pass the result to the second, then pass the result of the second to the third, and so on, finally returning the output of the last function. In other words, calling `pipe(foo, bar, baz)(1, 2, 3)` would be equivalent to calling `baz(bar(foo(1,2,3)))`.