# JavaScript for Front-End Exam – Punshing

You have been employed by a nightclub commercial company to create a web application which promotes nightclubs and their drinks, more importantly – their **Punshes**. Your task is to write the Front-End part of an application which extracts data from a Firebase database, and renders it into HTML.

There are a few simple tasks your employer would like you to do, to test your skills. Those little tasks will ultimately combine into the functionality you need for the application itself.

You will be given 4 **JavaScript files** which you will use for your solutions:

* **sample-data.js**
* **jquery-data.js**
* **events.js**
* **app.js**

## Rendering Sample Data – 20 pts

Write **2 JavaScript functions** which receive an **associative array** (**object**), and **render** the **data** from it.

The **object** looks like this:

punshes: {

0: { some punsh object },

1: { some punsh object },

. . .

}

And the punsh object looks like this:

0: {

name: {punshName},

type: {punshType},

contents: {punshContents},

description: {punshDescription}

}

The **properties** of the punshes object are:

* punshName is a **simple string**
* punshType is a **simple string**
* punshContents is a **simple string**
* punshDescription is a **simple string**

All of the objects will have **ABSOLUTELY ALL OF THEIR PROPERTIES**. There will be **NO** objects with **missing properties**.

Now that you are familiar with all the data, you should render it. The rendering is not complicated, just print it on the console with console.log()...

You will have to write **2 functions** which produce **2 different outputs** and receive different parameters. Use sample-data.js file for your solutions.

### renderAllPunshes() function – 10 pts

Write a JavaScript function - renderAllPunshes() which accepts the punshes object, and renders **ONLY** the **NAMES** of all punshes inside.

The output format is:

{punsh1name}

{punsh2name}

. . .

### renderSinglePunsh() function – 10 pts

Write a JavaScript function - renderSinglePunsh() which accepts the whole punshes object and a punshName, and renders all properties of the punsh with that **name**.

The output format is:

{punshName}

Type: {punshType}

Contents: {punshContents}

Description: {punshDescription}

See the examples below for more information.

### Example

|  |
| --- |
| sample-data.js |
| **let *punshes*** = {  0: {  **name**: **"Punsh - The American Pie"**,  **type**: **"Strong"**,  **contents**: **"Some Apple Pie, Whiskey, Vodka, Orange Juice and some other things..."**,  **description**: **"By original recipe from the American Pie franchise, this punsh includes everything you would want in a college/university party."**  },  1: {  **name**: **"Brendy Punsh"**,  **type**: **"Medium"**,  **contents**: **"Brendy, Apple Juice, Ice, Avocado Juice, some other strange fruits..."**,  **description**: **"The casual Brendy Punsh, quite expensive, nothing interesting here..."**  },  2: {  **name**: **"Just Punsh it"**,  **type**: **"Sweet"**,  **contents**: **"Very Little Vodka, Orange Juice, Apple Juice, Banana Juice, Ice."**,  **description**: **"A very comfortable taste for the lovers of weakly alchoholic drinks. The Just Pinsh It punsh is a sweet multi-vitamol drink, which cannot drunk you."**  }  };    *renderAllPunshes*(***punshes***);  *renderSinglePunsh*(***punshes***, **"Punsh - The American Pie"**); |

### Output

|  |  |
| --- | --- |
| renderAllPunshes() | renderSinglePunsh() |
| Punsh - The American Pie  Brendy Punsh  Just Punsh it | Punsh - The American Pie  Type: Strong  Contents: Some Apple Pie, Whiskey, Vodka, Orange Juice and some other things...  Description: By original recipe from the American Pie franchise, this punsh includes everything you would want in a college/university party. |

|  |
| --- |
| sample-data.js |
| **function** *renderAllPunshes*(punshes) {  *//****TODO: Implement me ...***  }  **function** *renderSinglePunsh*(punshes, punshName) {  *//****TODO: Implement me ...***  } |

## Rendering Data in HTML – 25 pts

Now that your employer knows that you are good in JavaScript, he wants to see some DOM manipulation skills. You will be given an HTML which will work as a template for you.

In order for the HTML to look tidy and well formed, CSS needs to be applied, which suggests some specific element hierarchy and classes. Check the template and make sure your function renders the object correctly. You can see input and output below.

Create your functions for this task inside the jquery-data.js file, and implement them so that they render the data inside the jquery-data.html file.

You already have some functions which render some data, you can probably use at least 50% of the code you used in the previous problem. Check in your solution from the previous problem, what can be **REUSED** in this problem.

### Render Punshes in HTML – 10 pts

You will be given a punshes object like the one in the **previous task**. **Render** the **names** of the **punshes**, and render them in the .navbar element, like they are **presented** in the **output HTML**.

#### Example

|  |
| --- |
| jquery-data.js |
| **let *punshes*** = {  0: {  **name**: **"Punsh - The American Pie"**,  **type**: **"Strong"**,  **contents**: **"Some Apple Pie, Whiskey, Vodka, Orange Juice and some other things..."**,  **description**: **"By original recipe from the American Pie franchise, this punsh includes everything you would want in a college/university party."** },  1: {  **name**: **"Brendy Punsh"**,  **type**: **"Medium"**,  **contents**: **"Brendy, Apple Juice, Ice, Avocado Juice, some other strange fruits..."**,  **description**: **"The casual Brendy Punsh, quite expensive, nothing interesting here..."** },  2: {  **name**: **"Just Punsh it"**,  **type**: **"Sweet"**,  **contents**: **"Very Little Vodka, Orange Juice, Apple Juice, Banana Juice, Ice."**,  **description**: **"A very comfortable taste for the lovers of weakly alchoholic drinks. The Just Pinsh It punsh is a sweet multi-vitamol drink, which cannot drunk you."** } };  *renderAllPunshesInHTML*(***punshes***); |

#### Input

|  |
| --- |
| jquery-data.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  </**div**>  <**div class="content"**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="jquery-data.js"**></**script**> </**body**> </**html**> |

#### Output

|  |
| --- |
| jquery-data.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  <**div class="navbar-items"**>  <**div class="navbar-item"**>  <**h4**>Punsh – The American Pie</**h4**>  </**div**>  <**div class="navbar-item"**>  <**h4**>Brendy Punsh</**h4**>  </**div**>  <**div class="navbar-item"**>  <**h4**>Just Punsh It</**h4**>  </**div**>  </**div**>  </**div**>  <**div class="content"**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="jquery-data.js"**></**script**> </**body**> </**html**> |

If you do everything correctly, you should see this:



### Render Single Punsh in HTML – 15 pts

You will be given a single punsh object like the one in the **previous task**. Render the name of the punsh in the .content-heading element, and **render** the **properties** of the **punsh**, in the .content element, like they are **presented** in the **output HTML**.

**NOTE:** The .content-heading element **SHOULD HAVE** a **CURSOR POINTER**.

#### Example

|  |
| --- |
| jquery-data.js |
| **let *punshes*** = {  0: {  **name**: **"Punsh - The American Pie"**,  **type**: **"Strong"**,  **contents**: **"Some Apple Pie, Whiskey, Vodka, Orange Juice and some other things..."**,  **description**: **"By original recipe from the American Pie franchise, this punsh includes everything you would want in a college/university party."** },  1: {  **name**: **"Brendy Punsh"**,  **type**: **"Medium"**,  **contents**: **"Brendy, Apple Juice, Ice, Avocado Juice, some other strange fruits..."**,  **description**: **"The casual Brendy Punsh, quite expensive, nothing interesting here..."** },  2: {  **name**: **"Just Punsh it"**,  **type**: **"Sweet"**,  **contents**: **"Very Little Vodka, Orange Juice, Apple Juice, Banana Juice, Ice."**,  **description**: **"A very comfortable taste for the lovers of weakly alchoholic drinks. The Just Pinsh It punsh is a sweet multi-vitamol drink, which cannot drunk you."** } };  *renderSinglePunshInHTML*(***punshes***, **"Punsh – The American Pie"**); |

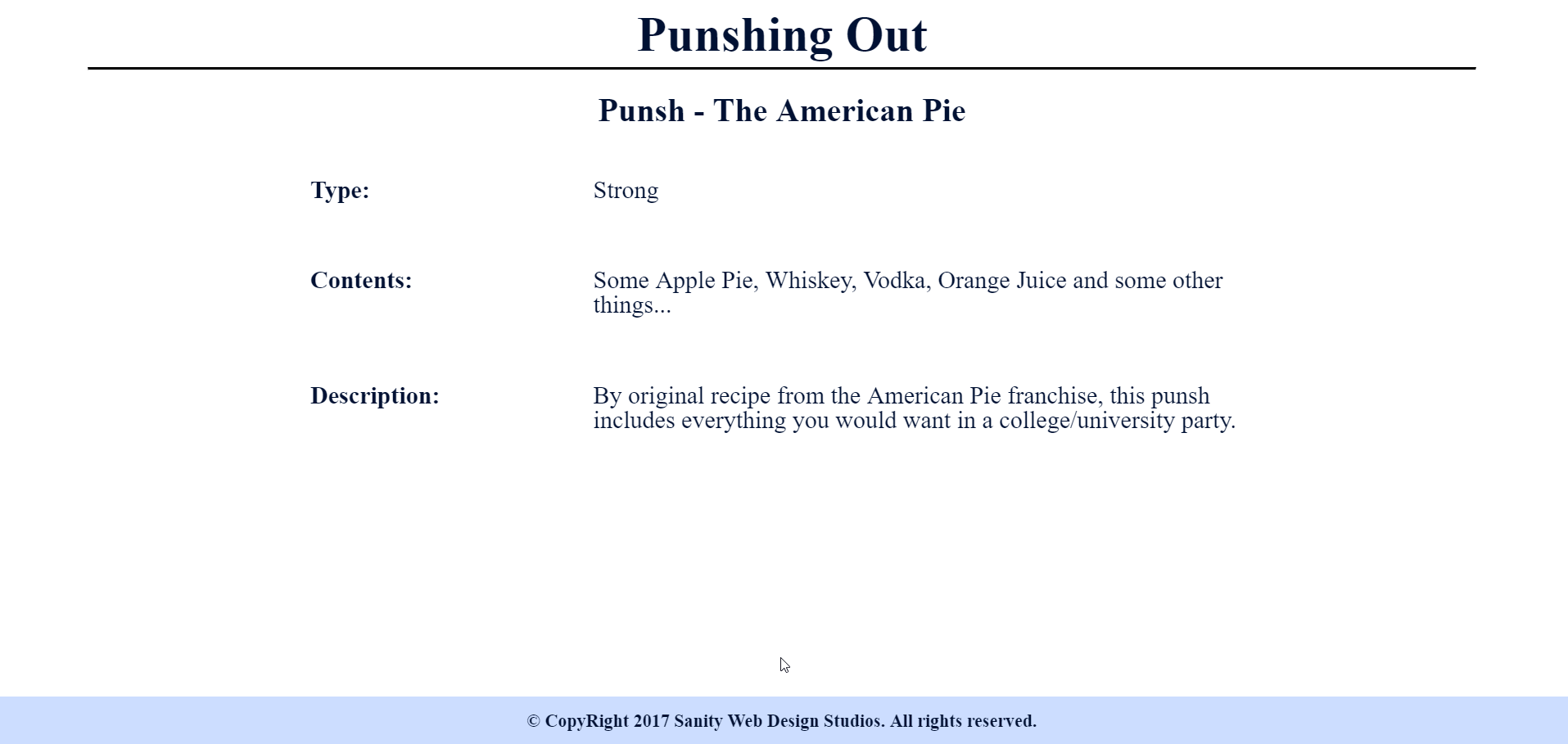
#### Input

|  |
| --- |
| jquery-data.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  </**div**>  <**div class="content"**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="jquery-data.js"**></**script**> </**body**> </**html**> |

#### Output

|  |
| --- |
| jquery-data.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  </**div**>  <**div class="content"**>  <**div class="content-header"**>  <**div class="content-heading"**>Punsh - The American Pie</**div**>  </**div**>  <**div class="punsh-data"**>  <**div class="punsh-type"**>  <**label**>Type: </**label**>  <**h6**>Strong</**h6**>  </**div**>  <**div class="punsh-contents"**>  <**label**>Contents: </**label**>  <**p**>  Some Apple Pie, Whiskey, Vodka, Orange Juice and some other things...  </**p**>  </**div**>  <**div class="punsh-description"**>  <**label**>Description: </**label**>  <**p**>  By original recipe from the American Pie franchise, this punsh includes everything you would want in a college/university party.  </**p**>  </**div**>  </**div**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="jquery-data.js"**></**script**> </**body**> </**html**> |

If you do everything correctly, you should see this:

****

**JQuery** may be used in this problem. It is **ADVISED** that you use **JQuery**, but **native DOM manipulation** is also allowed.

|  |
| --- |
| jquery-data.js |
| **function** *renderAllPunshesInHTML*(punshes) {  *//****TODO: Implement me ...*** }  **function** *renderSinglePunshInHTML*(punshes, punshName) {  *//****TODO: Implement me ...*** } |

## Capture Click Events – 30 pts

Your employer has acknowledged your JavaScript skills. But now he wants you to do some tasks of handling click events. Write your code in events.js.

### Punsh Click Event – 15 pts

Render all punshes (**ONLY** the **PUNSH NAMES**) by **REUSING** the code from the **previous task**, into events.html.

Your task is to **attach** a click event to each .navbar-item, which:

* **REMOVES** the whole .navbar-items element
* **RENDERS** the **PUNSH DATA** from the **CLICKED PUNSH**, into the .content element.
* **HINT**: Use the functionality from renderAllPunshesInHTML() and renderSinglePunshInHTML() from the **previous task**.

A Punsh-Click.GIF has been added to the resources, which displays this functionality.

### Back Click Event – 15 pts

Your task is to **attach** a click event to each .content-heading element, which **CLEARS** the .content element and renders all punsh **names** again.

A Back-Click.GIF has been added to the resources, which displays this functionality.

### Example

|  |
| --- |
| events.js |
| **let *punshes*** = {  0: {  **name**: **"One Punsh Man"**,  **type**: **"Strong"**,  **contents**: **"Very little Vodka, Very little Brendy, Very little Wine, Very little Whiskey, Very little Tequila and a lot of Watermelon Juice."**,  **description**: **"This punsh was discovered in an unknown house party, when there was almost no alcohol left, and the people had to combine a little from every alchohol. One Punsh of this is enough to knock you down for good, hence the name."** },  1: {  **name**: **"Wine Punsh"**,  **type**: **"Sweet"**,  **contents**: **"Wine, Apple Juice, Ice."**,  **description**: **"Casual Wine Punsh, for the ladies that love wine. It is not very strong, it is actually sweet. Sweet enough to make them sweetly drunk and playful."** },  2: {  **name**: **"Punsh Out"**,  **type**: **"Sweet"**,  **contents**: **"Champagne, Watermellon Juice, Ice."**,  **description**: **"This is a very sweet, almost non-alchoholic, punsh. Very suitable for ladies, which want to keep their manners but have a little fun at a party."** } };  attachPunshesEvents(***punshes***); *//pass the punshes object* |

|  |
| --- |
| events.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  </**div**>  <**div class="content"**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="events.js"**></**script**> </**body**> </**html**> |

|  |
| --- |
| events.js |
| **function** *attachPunshesEvents*(punshes) {  *//****TODO: Implement me...*** }  *//****NOTE: You might need the punshes in order for them to be rendered***  **function** *attachBackEvents*(punshes) {  *//****TODO: Implement me...*** } |

## Obtaining Data with AJAX – 25 pts

Now that your employer is sure of your skills, the time has come for you to combine what you have done in the previous tasks into one completed application.

You have been tasked to extract data about **punshes** from a Firebase database, and render it into the **HTML** into a specific format.

You see where we are going? All the tasks you’ve done so far, combine in order to implement a complete application. Check your previous code, and **REUSE EVERYTHING POSSIBLE**.

* **1st**, send a **GET** request to extract all punshes, but **RENDER** **ONLY** the **PUNSH NAMES**. Render them in the .navbar-items element. Then attach their click events.
* **2nd**, when a punsh is **clicked**, send **another GET request** to **LOAD** the punsh’s data. **Render** the **data** in   
  a .content element. Also **REMOVE** the .navbar-items, in the whole process. Attach the back event then.
* **3rd**, when the back event is triggered (the .content-heading is clicked) you should **CLEAR** the .content element, and **repeat** the **1st step**.

The **following URL’s** are for the **Firebase data**:

* **All Punshes**
  + https://punsh-master.firebaseio.com/data/punshes.json

**REUSE all functionality** from the **previous tasks**.

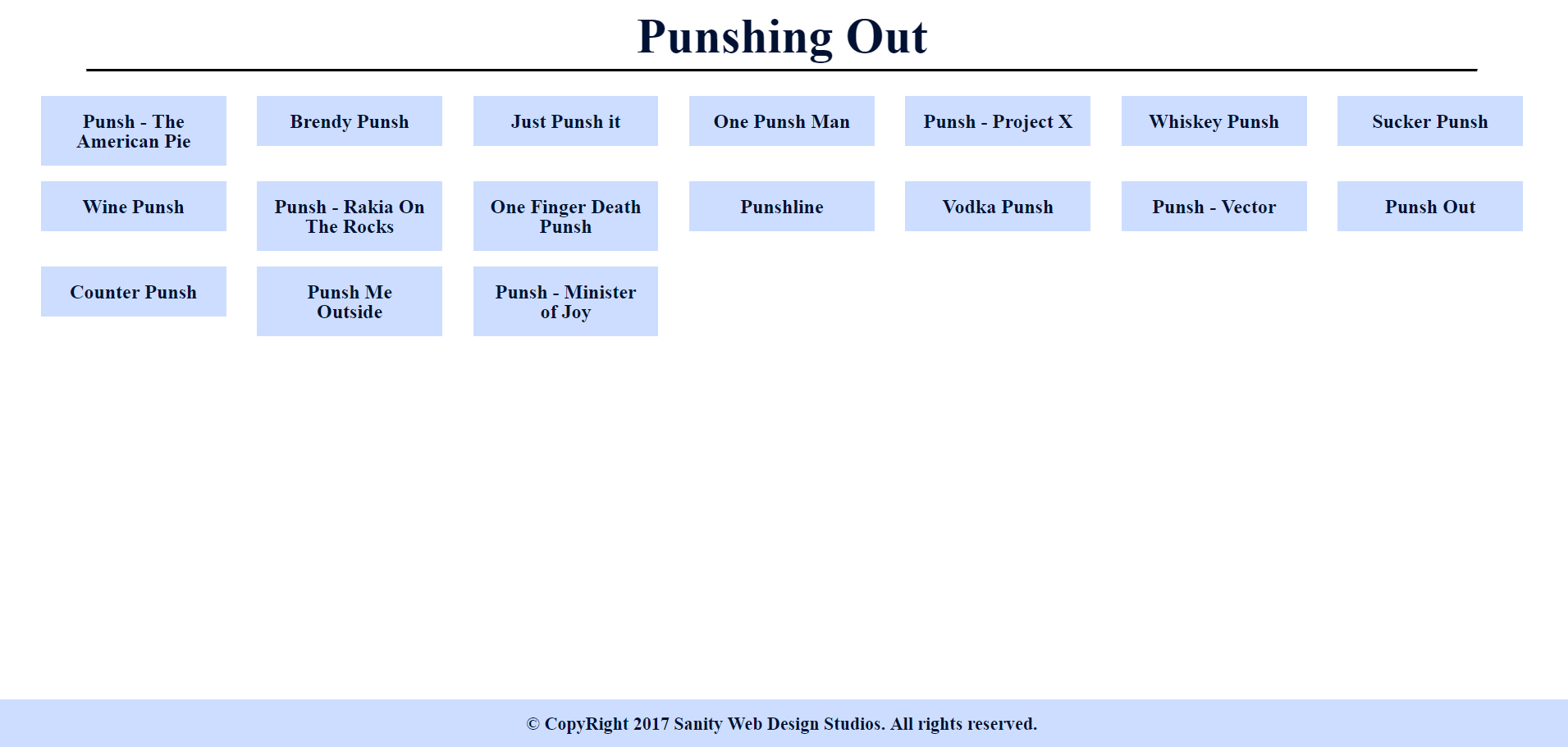
Write your code in the app.js file.

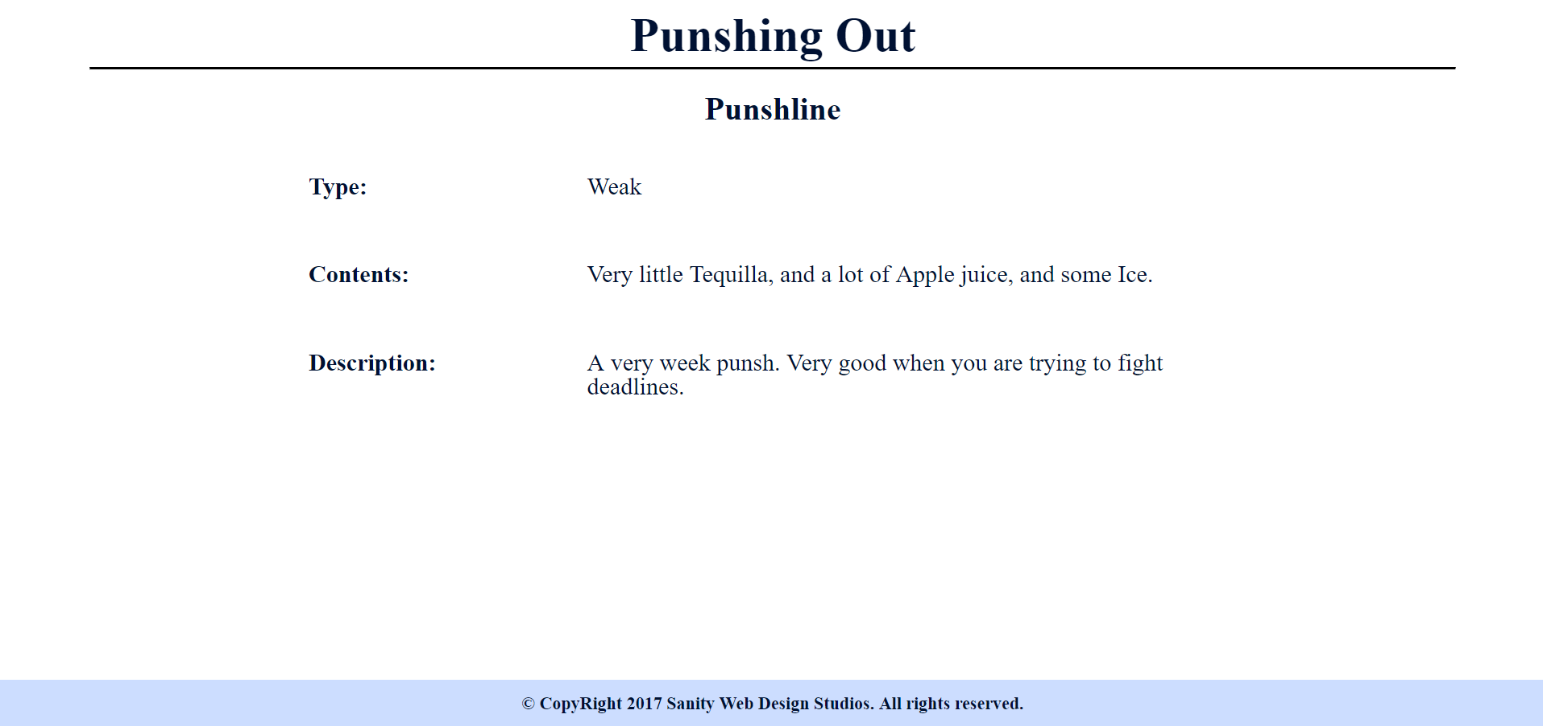
You will see several **screenshots** below which **describe** the **functionality**.

Here is the final HTML file you will need.

|  |
| --- |
| index.html |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>Punshing Out</**title**>  <**link href="reset.css" rel="stylesheet" type="text/css"**>  <**link href="style.css" rel="stylesheet" type="text/css"**>  <**script src="https://code.jquery.com/jquery-3.2.1.js"**></**script**> </**head**> <**body**> <**div class="container"**>  <**div class="navbar"**>  <**div class="navbar-header"**>  <**div class="navbar-heading"**>  Punshing Out  </**div**>  </**div**>  <**hr class="section-divider"** />  </**div**>  <**div class="content"**>  </**div**>  <**div class="footer"**>  <**div class="footer-caption"**>  <**h5**>**&copy;** CopyRight 2017 Sanity Web Design Studios. All rights reserved.</**h5**>  </**div**>  </**div**> </**div**> <**script src="app.js"**></**script**> </**body**> </**html**> |

### Screenshots





You will also be given a **GIF** in the **resources** which shows you the whole process.

Your application will be checked thoroughly for mistakes and errors in the code. Make sure you implement everything correctly. Check your code at least **3 TIMES** before submitting it to ensure that everything is correct. Sometimes just because it looks correct, does not mean it is correct.