

### Mini Challenge #3: Who is the Best Dancer from the Phantom Thieves?

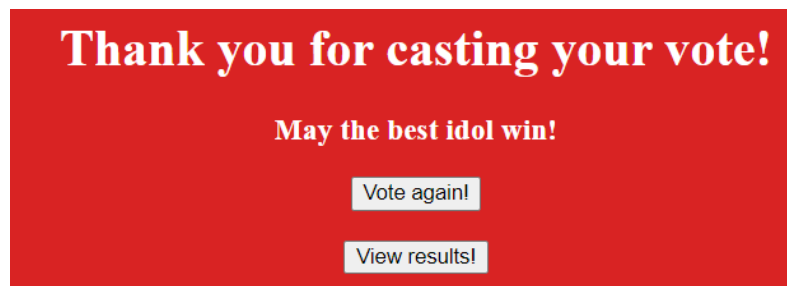
For this mini-challenge, you are going to develop a simple web application that will allow fans of the phantom thieves determine who the best dancer amongst them is. The reason is because the winner will be invited to perform in the Tokyo Dome alongside various artists.

However, everyone is preoccupied with their dance moves, thus their tech-genius (Futaba) is also busy practicing her moves. As a result, you have been commissioned to develop this application for them.



The specifications of the application is simple:

- 1.) The landing page presents the user with all members of the **Phantom Thieves**.
- 2.) A user will then click on an icon pertaining to a member and will be provided with the name and a preview of that member's dance number to help them decide.
- 3.) Once the user has decided on who they think has the best performance, they simply click a button located at the bottom of the landing page
- 4.) Clicking the button allows the user to cast their vote successfully, which will redirect users to a page called "confirm"



- 5.) At this point, users can either cast their vote again (to help their desired member become the center) or view the corresponding results showing the number of votes that have been cast for each member

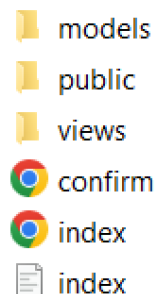


Image	Member Name	No. of Votes
	Ren Amamiya	6
	Ryuji Sakamoto	0
	Ann Takamaki	9
	Makoto Niijima	3

- 6.) Similarly to the confirmation page, the user is able to return to the landing page from the results page and cast another vote.
- 

## Task and To-do list:

You will be provided with the following files indicated in the screenshot and a brief description of the files below:



1. **models** – A folder containing a mongoDB/mongoose connection scheme and a Mongoose schema called Vote. You do NOT need to modify these files
2. **public** – Folder containing the image assets. DO NOT modify the contents

3. **views** – Folder containing a handlebar for results. You do NOT need to modify the file existing inside
4. **confirm.html** – File representing the .html file of a vote being confirmed. You do NOT need to modify this file
5. **index.html** – File representing the .html file of the landing page. You do NOT need to modify this file
6. **index.js** – Your main file to launch the application using Node. **You MUST complete this file.**

Your tasks for this mini-challenge are all inside index.js. Specifically, they are the following:

- 1.) Render index.html as the starting page of the application, and subsequently the home page **(10 pts)**.
- 2.) Save the votes made by the user into a MongoDB database using Mongoose **(10 pts)** and redirect users to a confirmation page indicating that they have voted **(10 pts)**
- 3.) Render confirm.html properly once a user's vote is confirmed **(10 pts)**
- 4.) Obtain the votes properly from the database **(10 pts)** and pass them correctly to a corresponding handlebar to be rendered properly **(10 pts)**.

Hints and Preparation:

- 1.) Instantiate your directory to become a node project via **npm init**.
- 2.) At the very least, these are the modules needed: **express**, **hbs**, and **mongoose** that have to be installed via npm
- 3.) HINT: Votes are cast using the following URL: /vote?q='<name>' where <name> = member's first name. Ex: /vote?q='ann' is a vote for **Ann Takamaki**.

Submission: Submit **ONLY** your index.js file