**Module 1 Assignment: Modules**

Brian Fredericksen

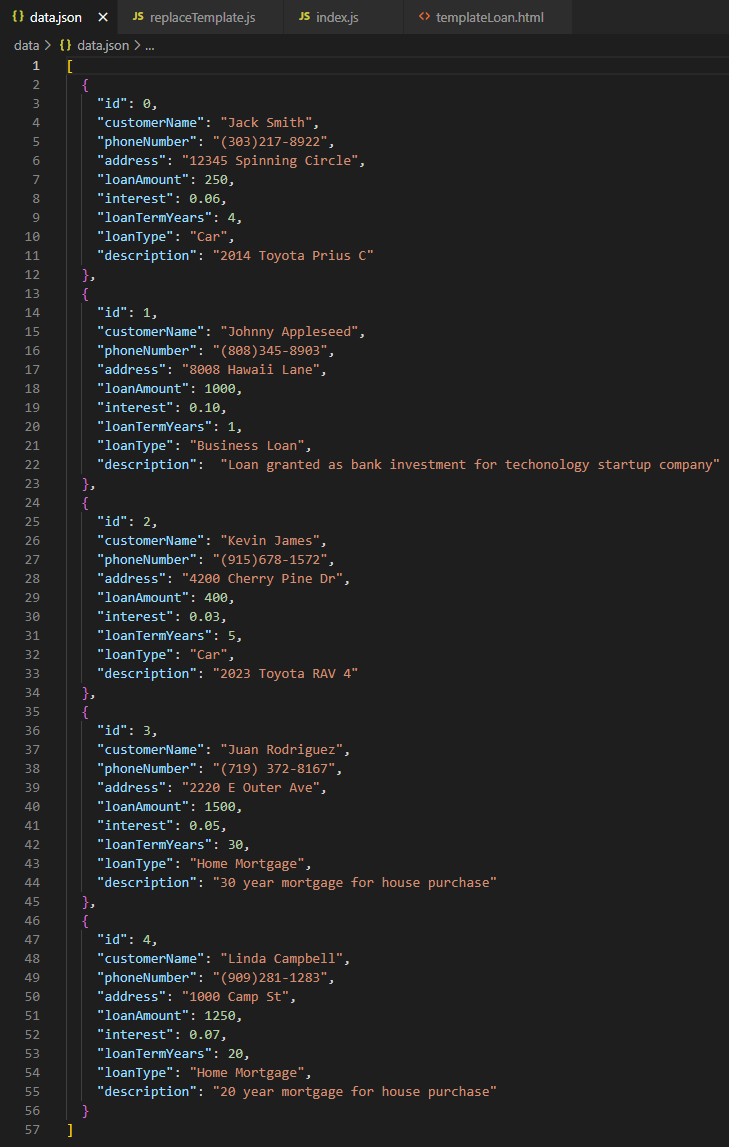
Ira A. Fulton Schools of Engineering, Arizona State University

IFT 458/598: Middleware Programming & Database Security

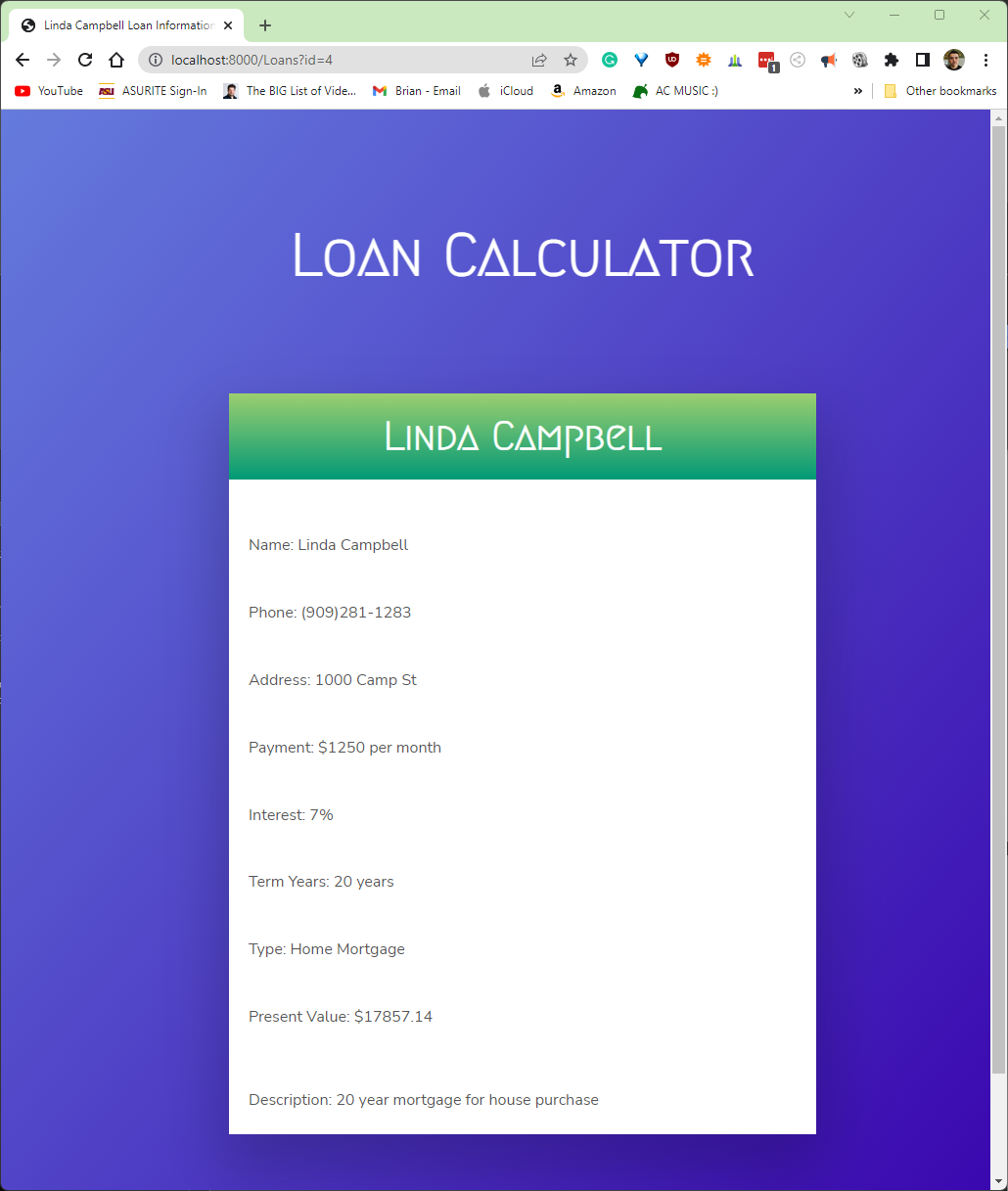
Professor Dinesh Sthapit

January 19, 2023

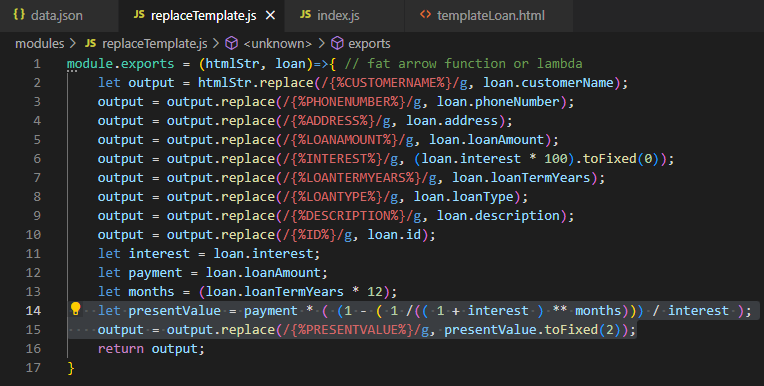
**Created 5 data elements based on the above specifications:**

****

**Added UI elements to display these data in the UI:**

****

**Created a computed field to calculate the total amount owed after the loanTermYears and display it in the UI:**

****

Note: I was a bit confused by the formula provided for calculating the loan amount, as I couldn’t seem to create it in a way that calculated the amount correctly. The formula provided seems to imply that what it’s calculating is present value, but from what I found from doing some research, the number this returns does not equate to the present value of the loan. As a result, I instead decided to go with the formula “P = PMT \* [1 – [ (1 / 1+r)^n] / r] where P = Present value of your annuity stream, PMT = Dollar amount of each payment, r = Discount or interest rate, n = Number of periods in which payments will be made” (Christian et al., 2023). Hopefully, while this is not the same formula as the one provided, its correct implementation here will demonstrate my ability to perform the task required by this assignment all the same.

**Research HTTP Status numbers and provide a brief description of each of the status values:**

* Information Responses (100-199):
  + 100 – Continue: Everything so far is okay, client should either continue request or ignore this response if the request is completed
  + 101 – Switching Protocols: Indicates which protocol a server is switching to after it receives an upgrade request
  + 102 – Processing: Server is processing the request but does not yet have a response available
  + 103 – Early Hints: Allows some resources to be preloaded while the server waits to return a complete response
* Successful Responses (200-299):
  + 200 – OK: Request successful
  + 201 – Created: Resource created as result of successful request
  + 202 – Accepted: Request is received but not yet fulfilled, likely to be handled by another server, process, or batch processing
  + 203 – Non-Authoritative Information: Metadata returned is from local or third-party copy not an exact match from origin server, usually used for mirrors or resource backups
  + 204 – No Content: No content can be sent in response to request
  + 205 – Reset Content: Document which sent request needs to be reset
  + 206 – Partial Content: Range header request only part of resource in the request
  + 207 – Multi-Status: Response designed for multiple resource that may need to respond with multiple status codes
  + 208 – Already Reported: Part of WebDAV, used to avoid repeating multiple responses for the resources in the same collection
  + 226 – IM Used: Response to a GET request that involves instance-manipulation
* Redirection Messages (300-399):
  + 300 – Multiple Choices: More than one response is possible, and the user is meant to pick one
  + 301 – Moved Permanently: Resource has moved to a new URL, which is provided by the response
  + 302 – Found: Resource has been moved to a new URL *temporarily*, but this URL should continue to be used by client
  + 303 – See Other: Requested resource should be found at another URL with a GET request
  + 304 – Not Modified: Used when response has not been modified since last access and the cached version of the response can be used
  + 305 – Use Proxy: Request response may only be accessed by a proxy (no longer used due to security concerns)
  + 306 – unused: Response code is no longer used for anything and is simply reserved
  + 307 – Temporary Redirect: Similar to response 302, but same HTTP method must be used in the next request (i.e. a GET request must be used for the second request if it was used for the first request)
  + 308 – Permanent Redirect: Similar to response 301, but same HTTP method must be used in the next request (i.e. a GET request must be used for the second request if it was used for the first request)
* Client Error Responses
  + 400 – Bad Request: Request can or will not be processed by the server due to a client-side error
  + 401 – Unauthorized: Client not authenticated to receive a response for their request as their identity cannot be confirmed
  + 402 – Payment Required: Reserved for future use, intended for use with digital payment systems
  + 403 – Forbidden: Client not authenticated to receive a response for their request as they’ve been identified as not having authorization
  + 404 – Not Found: Requested resource cannot be found, typically the result of an unrecognized URL when accessed in a browser, but occasionally appears instead of 403 when the server does not want the client to know said resource exists
  + 405 – Method Not Allowed: Method of request is known by the server but not supported with the requested resource
  + 406 – Not Acceptable: No content can be found that matches the criteria of the request
  + 407 – Proxy Authentication Required: Client not authenticated to receive a response for their request as authentication for said resource requires a proxy
  + 408 – Request Timeout: An idle or unused connection to the server has been shut down
  + 409 – Conflict: Request conflicts with the server in its current state
  + 410 – Gone: Requested content has been deleted from the server and no forwarding address may be provided
  + 411 – Length Required: Request did not contain a required Content-Length header
  + 412 – Precondition Failed: Server does not meet preconditions set by the request headers
  + 413 – Payload Too Large: Requested content exceeds limits set by server
  + 414 – URI Too Long: URI in request is longer than the server will interpret
  + 415 – Unsupported Media Type: Data requested in a media format that is not supported by the server
  + 416 – Range Not Satisfiable: Request has a Range header that cannot be fulfilled
  + 417 – Expectation Failed: Expect header in the request cannot be fulfilled by the server
  + 418 – I’m a teapot: Joke response that is occasionally used by servers for requests they don’t want to handle
  + 421 – Misdirected Request: Requested server cannot produce response for the given request
  + 422 – Unprocessable Entity: Semantic errors in request prevent the server from fulfilling it
  + 423 – Locked: Requested resource is locked
  + 424 – Failed Dependency: Request failed because of failure by a previous request
  + 425 – Too Early: Server refuses to process a request that it believes it received too recently
  + 426 – Upgrade Required: Request denied with current protocol but may be accepted if a different, upgraded protocol is used
  + 428 – Precondition Required: Request received that lacks specification of conditions that is required to fulfill request
  + 429 – Too Many Requests: To many requests have been sent by the client in a given timeframe
  + 431 - Request Header Fields Too Large: Request denied because some headers are too large
  + 451 – Unavailable For Legal Reasons: Resource is not able to access by the client legally, usually because of government censorship
* Server Error Responses (500-599):
  + 500 – Internal Server Error: Server experiences error it doesn’t understand to handle
  + 501 – Not Implemented: Request method not supported by server
  + 502 – Bad Gateway: Server is acting as a gateway to the response and received a response that was invalid
  + 503 – Service Unavailable: Server not ready to handle the request, usually as a result of being down for maintenance or overloaded
  + 504 – Gateway Timeout: Server is acting as a gateway and does not receive a response in time
  + 505 – HTTP Version Not Supported: Request received used a version of HTTP that the server does not support
  + 506 – Variant Also Negotiates: Internal configuration error has occurred where the resource is meant to negotiate what content is returned and therefore hits a dead end
  + 507 – Insufficient Storage: Not enough storage space on the server for the content represented by the request
  + 508 – Loop Detected: Request lead to an infinite loop of requests
  + 510 – Not Extended: Request requires continued extensions in order to be fulfilled
  + 511 – Network Authentication Required: Client is attempting to gain network access without proper authentication

**Research HTTP Response headers and provide a brief description of each:**

* Feature Page Elements:
  + Name: Name found in the name-value pair
  + Value: Value found in the name-value pair
  + Entry Type: Indicates whether the item is local or inherited, meaning from the current configuration file or parent configuration file, respectively
* Actions Pane Elements:
  + Add: Opens dialog box for adding headers
  + Set Common Headers: Opens dialog box for configuring keep-alive and Web content expiration headers
  + Edit: Opens dialog box for editing a header
  + Remove: Removes selected item from feature page list
* Add or Edit Custom HTTP Response Header Dialog Box:
  + Name: Name found in the name-value pair
  + Value: Value found in the name-value pair
* Set Common HTTP Response Headers Dialog Box:
  + Enable HTTP keep-alive: Option for keeping a connection open between client and server across multiple different requests
  + Expire Web content: Option to configure when client’s cache of a web page should be cleared

**References:**

Christian, R. (2023, January 10). *Present value of an annuity: How to calculate & examples*. Annuity.org. Retrieved January 22, 2023, from https://www.annuity.org/selling-payments/present-value/

Mozilla Corporation. (2023, January 17). *HTTP response status codes - HTTP: MDN*. HTTP | MDN. Retrieved January 22, 2023, from https://developer.mozilla.org/en-US/docs/Web/HTTP/Status

Microsoft Archived Docs. (2016, August 31). *HTTP response headers*. Microsoft Learn. Retrieved January 22, 2023, from https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-R2-and-2012/hh831707(v=ws.11)