/\*\*

 \* Journal Design

 \* Tracy Mann

 \* CSE210

 \* Bro. Parrish

 \* Section A8

 \* 4/29/2023

 \* v1.1

 \*

 \* What does the program do?

 \*      For this assignment you will write a program to help people record the events

 \*   of their day by supplying prompts and then saving their responses along with the

 \*   question and the date to a file.

 \*

 \*   Functional Requirements

 \*      This program must contain the following features:

 \*    1)  Write a new entry - Show the user a random prompt (from a list that you create),

 \*    and save their response, the prompt, and the date as an Entry.

 \*    2)  Display the journal - Iterate through all entries in the journal and display

 \*    them to the screen.

 \*    3)  Save the journal to a file - CipheredPrompt the user for a filename and then save the

 \*    current journal (the complete list of entries) to that file location.

 \*    4)  Load the journal from a file - CipheredPrompt the user for a filename and then load the

 \*    journal (a complete list of entries) from that file. This should replace any entries

 \*    currently stored the journal.

 \*    5)  Provide a menu that allows the user choose these options

 \*    6)  Your list of prompts must contain at least five different prompts. Make sure to

 \*    add your own prompts to the list, but the following are examples to help get you started:

 \*          Who was the most interesting person I interacted with today?

 \*          What was the best part of my day?

 \*          How did I see the hand of the Lord in my life today?

 \*          What was the strongest emotion I felt today?

 \*          If I had one thing I could do over today, what would it be?

 \*    7)  Your interface should generally follow the pattern shown in the class video demo.

 \*   Design Requirements

 \*      In addition, your program must:

 \*    1)  Contain classes for the major components in the program.

 \*    2)  Contain at least two classes in addition to the Program class.

 \*    3)  Demonstrate the principle of abstraction by using member variables and methods

 \*    appropriately.

 \*  Stretch Requirements

 \*    1)  Add option to save and load JSON format.

 \*    2)  Read configuration for the application in JSON format from config file.

 \*    3)  Add checks for file extension assitance:  .csv for CSV file and .json for JSON file.

 \*    4)  Add counter for prompt questions used.

 \*    5)  Add last date used for prompt questions.

 \*    6)  Add distribution balancer for random selection of prompt questions.

 \*    7)  Add freeform prompt when all prompts have been used in the same day.

 \*    8)  Ensure csv files are safe for MS Excell loading.

 \*    9)  Change data store to a sqlite DB instead of list of enties in memory and json file for prompts.

 \*   10)  Added encryption to both the database and file storage methods.  Maintained encryption on the data in the database, and on the prompts configuration file.

 \*

 \* What user inputs does it have?

 \*      1)  menu prompt response.

 \*      2)  entry prompt response.

 \*      3)  save/load filename prompt response.

 \*      4)  save/load file format prompt response.

 \* What output does it produce?

 \*      1)  display menu:

 \*              1)  Add Journal entry.

 \*              2)  Display journal.

 \*              3)  Load journal.

 \*              4)  Save journal.

 \*              5)  Exit.

 \*              >

 \*      2)  display random entry prompt.

 \*              Who was the most interesting person I interacted with today?

 \*              >

 \*              What was the best part of my day?

 \*              >

 \*              How did I see the hand of the Lord in my life today?

 \*              >

 \*              What was the strongest emotion I felt today?

 \*              >

 \*              If I had one thing I could do over today, what would it be?

 \*              >

 \*              What did you do today to serve someone?

 \*              >

 \*              What did you ponder over in your scriptures today?

 \*              >

 \*              What Christ-like attribute did you emulate today and how?

 \*              >

 \*              How were you helped today and by who?

 \*              >

 \*              What outside event did you witness or ponder today?

 \*              >

 \*              What did you do today, that was enjoyable?

 \*              >

 \*              What did you learn today?

 \*              >

 \*              What did you improve in your life today?

 \*              >

 \*              What happened today that you didn't like and how could you manage it better?

 \*              >

 \*              What would you like to write in your journal?

 \*      3)  request filename prompt.

 \*              please enter the filename:

 \*              >

 \*      4)  file format menu:

 \*              1)  CSV.

 \*              2)  JSON.

 \*              >

 \*      5)  report saving file.

 \*              saving to file...

 \*      6)  report loading file.

 \*              loading from file...

 \*      7)  report unable to read csv.

 \*              Unable to read file as CSV.

 \*      8)  report unable to read json.

 \*              Unable to read file as JSON.

 \*      9)  display journal entries.

 \*              Journal:

 \*

 \*                  {date} prompt: {prompt}

 \*                  {response}

 \*

 \*                  {date} prompt: {prompt}

 \*                  {response}

 \*                  ...

 \*      10)  exit message

 \*              Thank you for using the Journal Program, hope to see you tomorrow.

 \* How does the program end?

 \*      The user will select the exit option from the menu.

 \*

 \* Class Diagrams:

 \*      Application:

 \*          Responsibilty:

 \*              to hold the journal information and manage operations performed on the information.

 \*          State:

 \*              journal Journal

 \*              is running Boolean

 \*          Behaviors:

 \*              empty constructor

 \*              journal accessors

 \*              is running accessors

 \*              database accessors

 \*              encryption accessors

 \*              Initialization Vector accessors

 \*              Secret Key accessors

 \*              File accessors

 \*              run application

 \*              display menu

 \*              read menu response

 \*              evaluate menu response into journal/application behavior

 \*              exit

 \*      Journal:

 \*          Resonsibility:

 \*              to add, hold and display journal entries.

 \*          State:

 \*              file data and tools

 \*              database data and tools

 \*          Behaviors:

 \*              empty constructor

 \*              File accessors

 \*              database accessors

 \*              encryption accessors

 \*              prompt for entry

 \*              read response

 \*              add a new journal entry

 \*              display all journal entries

 \*      Prompt:

 \*          Responsibilty:

 \*              to hold encrypted information about the available prompts for jornal entries.

 \*          State:

 \*              value string

 \*              timesUsed string

 \*              lastUsed string

 \*          Behaviors:

 \*              constructor for encrypted string values

 \*              constructor for open data type values

 \*              constructor for open string values

 \*              value accessors

 \*              set open value

 \*              get open value

 \*              last used accessors

 \*              set open last used string

 \*              get open last used string

 \*              set last used datetime

 \*              get last used datetime

 \*              times used accessors

 \*              set open times used string

 \*              get open times used string

 \*              set open times used integer

 \*              get open times used integer

 \*              display prompt

 \*              json accessors

 \*              conversion to open prompt object

 \*      OpenPrompt:

 \*          Responsibilty:

 \*              to hold non-encrypted information about the available prompts for jornal entries.  this is an internal object to the prompt object, used for building json non encrypted data.

 \*          State:

 \*              value string

 \*              timesUsed int

 \*              lastUsed DateTime

 \*          Behaviors:

 \*              constructor for open string values

 \*              constructor for open data type values

 \*              constructor for encrypted string values

 \*              value accessors

 \*              set ciphered value

 \*              get ciphered value

 \*              last used accessors

 \*              set ciphered last used string

 \*              get ciphered last used string

 \*              set last used string

 \*              get last used string

 \*              times used accessors

 \*              set ciphered times used string

 \*              get ciphered times used string

 \*              set open times used string

 \*              get open times used string

 \*              json accessors

 \*              conversion to prompt object

 \*      Entry:

 \*          Resonsibility:

 \*              to hold and display encrypted information about an individual entry event.

 \*          State:

 \*              date string

 \*              prompt Prompt

 \*              response String

 \*          Behaviors:

 \*              constructor for encrypted string values

 \*              constructor for open data type values

 \*              constructor for open string values

 \*              date accessors

 \*              set open date

 \*              get open date

 \*              set open date string

 \*              get open date string

 \*              prompt accessors

 \*              prompt value accessor

 \*              set open prompt value

 \*              get open prompt value

 \*              prompt times used accessor

 \*              set prompt times used integer

 \*              get prompt times used integer

 \*              set prompt open times used string

 \*              get prompt open times used string

 \*              prompt last used accessor

 \*              set prompt last used date

 \*              get prompt last used date

 \*              set prompt open last used string

 \*              get prompt open last used string

 \*              response accessors

 \*              set open response

 \*              get open response

 \*              display entry

 \*              get csv

 \*              parse csv

 \*              json accessors

 \*              conversion to open entry object

 \*      OpenEntry:

 \*          Resonsibility:

 \*              to hold and display non-encrypted information about an individual entry event..  this is an internal object to the prompt object, used for building json non encrypted data.

 \*          State:

 \*              date datetime

 \*              prompt OpenPrompt

 \*              response String

 \*          Behaviors:

 \*              constructor for open string values

 \*              constructor for open data type values

 \*              constructor for encrypted string values

 \*              date accessors

 \*              set open date string

 \*              get open date string

 \*              set ciphered date

 \*              get ciphered date

 \*              open prompt accessors

 \*              open prompt value accessor

 \*              set open prompt ciphered value

 \*              get open prompt ciphered value

 \*              open prompt times used accessor

 \*              set open prompt times used string

 \*              get open prompt times used string

 \*              set open prompt ciphered times used

 \*              get open prompt ciphered times used

 \*              open prompt last used accessor

 \*              set open prompt last used string

 \*              get open prompt last used string

 \*              set open prompt ciphered last used string

 \*              get open prompt ciphered last used string

 \*              response accessors

 \*              set ciphered response

 \*              get ciphered response

 \*              json accessors

 \*              conversion to entry object

 \*      Journalfile:

 \*          Resonsibility:

 \*              to hold and manage file operations for the files used by the journal object.

 \*          State:

 \*              Prompt Data File string

 \*              Does Prompt Data file Exist Boolean

 \*              Json prompt file is intialized Boolean

 \*          Behaviors:

 \*              empty constructor

 \*              Prompt Data File accessors

 \*              Does Prompt Data file Exist accessors

 \*              Json prompt file is intialized accessors

 \*              load entry prompts

 \*              update prompt data

 \*              prompt for filename

 \*              evaluate file format

 \*              prompt for file format

 \*              prompt for base filename

 \*              load journal entries

 \*              verify is csv data

 \*              verify is json data

 \*              save journal entries

 \*              get csv

 \*              set csv

 \*              get json

 \*              set json

 \*              prompt for aes encrypted response

 \*              get for aes encrypted response

 \*              encrypt file with RSA

 \*              decrypt file with RSA

 \*      JornalDatabaseConnection:

 \*          Resonsibility:

 \*              to interface and manage databse operations for the data used by the journal object.

 \*          State:

 \*              database filename

 \*              database read write mode

 \*              database cache policy

 \*              database use foreign key policy

 \*              database default timeout

 \*              database to be initialized

 \*              encryption object

 \*          Behaviors:

 \*              base constructor with values

 \*              database filename accessors

 \*              database read write mode accessors

 \*              database cache policy accessors

 \*              database use foreign key policy accessors

 \*              database default timeout accessors

 \*              database to be initialized accessors

 \*              encryption accessors

 \*              get database connection string

 \*              get database connection

 \*              get database query result object list

 \*              is datbase defined read only accessor

 \*              define database

 \*              Are database Prompts Defined read only accessor

 \*              define database Prompts

 \*              read database Prompts

 \*              update database Prompts

 \*              Get Entry Prompt

 \*              Add database Journal Entry

 \*              Read database Enties

 \*              Truncate database Enties

 \*      Encryption:

 \*          Resonsibility:

 \*              to interface and manage encryption operations for the data used by the journal object.

 \*          State:

 \*              aes block size

 \*              aes feedback size

 \*              aes key size

 \*              aes cipher mode

 \*              aes padding mode

 \*              aes legal block sizes

 \*              aes legal key sizes

 \*              rsa crypto service provider bit size

 \*              aes encryption object

 \*              rsa crypto service provider object

 \*          Behaviors:

 \*              constructor for all default encryption states

 \*              aes block size accessors

 \*              aes feedback size accessors

 \*              aes key size accessors

 \*              aes cipher mode accessors

 \*              aes padding mode accessors

 \*              aes legal block sizes accessors

 \*              aes legal key sizes accessors

 \*              rsa crypto service provider bit size accessors

 \*              aes encryption object accessors

 \*              rsa crypto service provider object accessors

 \*              set aes key and encryption state

 \*              encrypt sting with aes

 \*              decrypt string with aes

 \*              prompt for base filename to rsa encryption keys

 \*              read responses

 \*              display public and private key pair filenames

 \*              get new public and private key pair

 \*              get public key

 \*              get private key

 \*              get rsa parameters

 \*              get rsa key string

 \*              set rsa key

 \*              encrypt string with rsa

 \*              decrypt string with rsa

 \*              encrypt large string with rsa

 \*              decrypt large string with rsa

 \*

 \*      Notes:

 \*          I found the json format notes on stack overflow:  https://learn.microsoft.com/en-us/dotnet/standard/serialization/system-text-json/how-to?pivots=dotnet-7-0

 \*          file exists test:  https://learn.microsoft.com/en-us/dotnet/api/system.io.file.exists?redirectedfrom=MSDN&view=net-7.0#System\_IO\_File\_Exists\_System\_String\_

 \*          for loop:  https://www.w3schools.com/cs/cs\_for\_loop.php

 \*          date compare:  https://learn.microsoft.com/en-us/dotnet/api/system.datetime.compare?view=net-8.0

 \*          clean time from date:  https://stackoverflow.com/questions/1859248/how-to-change-time-in-datetime

 \*          I had added sqlite toolbox to my VS IDE and sqlite package to add the Microsoft.Data.Sqlite package.

 \*          I had used the micrsoft documentation for sqlite implementation.

 \*          I used my basic understanding of sql from work to build the db schema and sql queries/updates/inserts.

 \*          I got notes for aes encryption at https://learn.microsoft.com/en-us/dotnet/api/system.security.cryptography.aes?view=net-7.0

 \*          I got notes for rsa public/private key encryption at https://stackoverflow.com/questions/17128038/c-sharp-rsa-encryption-decryption-with-transmission

 \*          I got notes that i need to change how to work with the public/private key from this article:  https://social.msdn.microsoft.com/Forums/windowsapps/en-US/29503d96-95b9-4a51-a787-ee7ba5f1d3ae/systemsecuritycryptographycryptographicexception-bad-length-in-rsacryptoserviceprovider?forum=wpdevelop

 \*

 \*      Class Diagram:

 \*      //www.plantuml.com/plantuml/png/j5bVazks4N-\_Ju59NkJsh74zgPl99KufKNALa-vKK-TEs-bm879FXasH305U8hlttGkGM9802KbsRJ\_eoDtV\_dQnk\_X3-AzSq4AK--GYfz5xkWEoOb6HyUnUZ7qgw3ZuDM42oCSB8lz52UMSrA9fbgOG2PQb7nyl7YyQ--301Up70KHbmSHXF2qEkSXs1StV7WmQ3pYHfgtWTU3Lu02A1nR1kiWUM0n53xMc1Ts3W88Rcciu1Em3e13ISa-cB7yBnIgBeIjTqpXcwKwBPV0weZnFM4JLc2xgI9-IIPObG5CIClwwJ5DfTQKrlsHbaTA4XE\_gXrh-noxBw4ajGop-RUGF6OiR8O8q7s8mWWK6O5gtOWHoGGLOdbKjjvHtgaPGPD8rKhTXh4LeDqiZLJQbQXzDrewyWuRc4S7RWu1\_\_uSiKYOOJTY7AejlfD-iC0a74ThwFHmiFjcC1GZP7oP98-vROjxcB06YVapBIjA5tCiV0o6BOYMMyJoXXvLim4R1HI4BHru3ZGqvLdBsG9EI2gYK3PZdMSh1pE\_lJ4ZhnujEyv4\_F6sQr93\_-5-bS9Mt3sqhvgHHET2p24KuWunCXYfFE87S7HPYYrsPrHXzVdM-veI8ciUlC2NMHRRFXVuppufPAeh3I4lpwiz0Es6NZ2mQBuxrE0PuBf6XqDP33LorcUR05iD46t9dlX1bTPSfrm5ZKtMSwCRwg9Ov5ZsIfnXrk7vzkrflGj\_RUE5yiPoHdyWtUZ4TouxwvifUq-jXy\_0TpzAGfKnO\_OQjrNM7QTO1gk4DmRwjacQ3jAdgnvJkeKUDaVaPyCPKJcukxBL1C\_DVWjiRjHYQpMKh-ri38fSPZQjMq2RdhljeVvTBCRJ376xcIz9lM6HeEdkkis8lZxWAAhs4xQcmjyfBOYVMvG09XW4Jog69GbFAFM5po8UJne2V7SfMbe0i-3HuWymgdvPM5QolnksAFi0dUaNnAn3IiQCss58eWrXEd7kQM9KEZd6W8vRcfN0p3iIbXln\_1eOaNo0oeoCyu9eJ-Wig4hs2OPnLeQ6rNxMQFCwEWZh2FA4itS3lCahdK4mWzcgZsUe6lkctW0-6p-CUeeCwDfzsqMu5IdJYPD67OX1z7Qx\_JdhS69pUtDk4FwbsXnjlLUqGcc4SRzK6uJHPtVeLNEsk\_oZK1r01LC1Hcwat\_Ljbj2xOmodmNJ2RNolmfYYF8\_tPFDmiLhFRkut2ltY6ErkzZ7HsicwAITWM-khKOMXKCSNsDkHgJTSUwM-nVnpxOmDmxuqzobKME-XGgQ5x-UoccTBehOi7jJWTr8kRwOwhRON1Bhs6WulHH26pGXrXzk6UbedOi3rafN1Icp3a5lMhjSNq3WieKtgCvmkVOZ5YO-7t9ssx1PLqr4FRLmssNLCWYwb7j6HS\_7YxVIT1BskIlvTG76g9KYBJhqfyILWgr2bO9dMQ9TEiJ8NphBRW\_aGMWAKG3tgjYL-I6kDFHauchv1gVGxJZ0W-peb6hbMziganB\_KHZ0yJ8qYjSYTeZqQcpuLV9h9QhyhMEQb-Jb1UB0TbVFyumggK1DJlC6C5m3J95mQVbIMvrAUHoXDou8xQNvJnYxxZvGTISk3afTkYlvMwKTMGZe2oSvWo93hbhJUpVo0x478mkB7ZE8TsSZKxGsyMX9FbxVGw31R\_Khlidvv\_V-L4pcSpV-Aru2FGwza\_4VNypozwkFPAiOBZmQ0LZwUJQSyAhnmRixNd-ukRLzAoenglhwV1Nth6UBDPMovdhxnbP-Y1v4ZXjmOraecu9DNFioVdSE687Kmo2PVanNTFXZCc3vFXD5Y7ayL6cJz\_zjttUFRmb1SA7D-T5xnXnFEhtgOpQa9DO4UJIP95xwltdz8oIIw9gTPlg5I-ARwvFKl75QGtJTNOP8BGgUEWO4AK3PSFBh8vGBobn\_aGTIQb7hMJ3G5kehPl5RnvSr6QcFPnyB25hOti-Egcjq3ux7I0LLU6IxDBt1vi899WWawuG87R0S9EqMCXLNi6wmaJBiuEGE6tyW6FOLuTcNaiKm8z0r3kd08kXS7eHXsRMlPdkWm9hzhPm0N-9T5AZXVprK7iPxB5Mia5HirMBRx7cTBJl8U38NIj94C0kiENHjks-SzL6w82-xlMvFMBLfgVtxYi1TMRAe\_zBT\_xN1\_6HrKe49znWNNwMjYysLsNsuH5-bDLvc1DMTi4cp9FG9zRy2wWEadVm6zkG-F0NyEa\_cZFOFfwviZ\_IVRcVnSgcgwWSyTd-jD0wKj\_cPWqY6zQ6z5kGgF-4joh\_\_btE3O9NjUSPAdjvFpImInfiOEMBFYKY0ACoCca79qCxF52nuB7Mm-uBN9L6t7DcHBdRK\_HpbzRXJDuK6SK7S-bzLpJPzBsMaRTKSEL9LHVuFoEGsmh99DevVY5W-Mik9g48ROHDAnFJ6lNbTxN9\_MfW8qawD2oLBISSau-irhvZR8qbY4jwZ8RwjiSqWfoJaocjrEs5dgfHs66rqYj5FQ2HA0hJeUz5fzc68nAbS3bi5SUP425YyMT8cN5-pbgnQwsQhA4K6UQ70meFuDZEBH5wkngesdHMWbF6mzjgCUGt-8ugl-wG9cohYLQ-uwZmTDgnFegP9ErBoulZnSQR5sr6-AMvd\_5carE0p4CXQ\_2kk8GI5m1iKUYEs01ulveS1-U-Uh6hZF0Dh5VVfg-pnb5\_cOjYSVfyaU3ynD5YvDkn-QWaBgIQgH76m2jEkjd7G-30V\_1x-xNgwYlcDvNqQ4VP0uO6medNlH1\_YuNfj\_mn5FjDlHR1lDVFpIvm2-m1xdB9MSGkz1nF2qjCLI6DtpAz7uyAlEdbQ7HezYrzIZxmaXxMZuuouSiNYyyM7xfVSl9cqnFIrbaRcyg3sEKc65tDyN63ArvcSlMVQnb9NW6Q4s\_j3mwI44X731VELV6pFmLLVZwO\_YYolhSmf4R6Zc3PifNADGhD4iCB0oTj8P1SBT-WaJnJ22nNolMbC9z3s78Fa8c3fslr6jQmKvwOlzkKjL1MlZLoLlfN9ov0kEn4kU3mcpftdzLnJFnzFvnNBAtpvEgMbENY6\_6dEuGu0qHRQ479x0r4DVUwWEiBnG2DYy9ZcnBl0hRKWQwJIY38mGqfRv4YTbrNKte4hb1NtHxnHTJSnqKrgqz5kVPk5ebJ5RKRt9PSswokQv27fRXJAh6aTKwdqcZKx0FVQzIYu5NOp3\_4moitWl9\_cINTY0hjFwAhXfrYh15bVPrKhHAIELWEso\_Qta7gSKUezRTyUtvy\_Nb2m00

 \*/