

## Design Report for Project 1:

### Main.py

- Calls all additional classes
- Runs the main program
- Continues to run until q is selected
- chooseMainMenuOption()
  - choose q or a number to start functionality

### userInput.py

- Used to get correct user input for most functions
- Used in most classes

### person.py

- Used in functions to handle details about a person
- isValidGender(gender)
  - Takes their gender as a param
  - Checks for correct gender
- checkExists(conn, sin)
  - Take the database URL string (conn) and person's SIN as params
  - Checks if the user exists or not by using the COUNT feature
- register(conn, sin)
  - Takes database URL string as param conn
  - Takes person's SIN as param sin
  - If person does not exist, function to register person
  - To quit at anytime press control - d to create a keyboard interrupt error

### vehicle.py

- used in classes to handle vehicle details
- checkExists(conn, serial)
  - Takes database URL string as param conn
  - Takes serial # of vehicle as param serial
  - checks if serial number of vehicle exists
- hasOwner(conn, serial, sin)
  - Takes database URL string as param conn
  - Takes serial # of vehicle as param serial

- Takes sin of owner as param sin
- Check the owner of the vehicle
- hasPrimaryOwner(conn, serial, sin)
  - Takes database URL string as param conn
  - Takes vehicle serial # as param serial
  - Takes sin of primary owner as param Sin
  - Checks the vehicle for the primary owner of i

#### newVehReg.py

- Used to register a new vehicle
- setupVehicleReg(conn)
  - Called by main to start regiserty of a vehicl
- At any time can be canceled by keyboard inter
- ruption
- vehicleRegPart(conn)
  - Called by setupVehicle
  - Gets a unique serial #
  - Makes sure vehicle type exists
  - Gets all other information about the vehicle
- Returns to the serial number
- addOwnersPart(conn)
  - Called after vehicleRegPart
  - Takes the serial number returned by vehicleRe
- gPart as a param
  - Adds owners to vehicle
  - Can add primary as well as non-primary
- Note all functions can be canceled by keyboard in
- teruption

#### newDriver.py

- Used to enter in a new driver
- calls newDriverImpl
  - Gets the information needed to set up a licen
- ce
  - Makes sure licence does not already exists
  - Inputs a picture if possible

#### autoTran.py

- Used to enter an auto transaction

- Deletes old owner and adds the new owner
- performTransaction
  - Called by main to start the auto transaction
- performTransactionImpl
  - Called by performTransaction
  - Gets the information from the vehicle
  - Gets the Seller and the buyers SIN
  - Has checks for each to make sure they exist
  - Deletes the current owners
  - Add the buyers as the owners

#### violationRec.py

- Used to handle recording a ticket
- called by main
- Calls recordTicketImpl to record ticket
  - recordTicketImpl gathers the information needed to record ticket
  - Gets the serial number of vehicle
  - If input left blank after serial number, violator will be primary owner

#### searching.py

- handles option 5 searches
- serachOption
  - called by main to set up which search to use
- personalInfo
  - Called by searchOption to handle searching by licence or name
- personalByLicence
  - Called by personalInfo to handle searching by licence
- personalByName
  - Called by personalInfo to handle searching by name
- violationRecords
  - Called by searchOption to handle violation searches
- violationBySin
  - called by violationRecords to search by sin
- violationByLicence
  - called by violationRecrods to search by Licence

- vehicleHist
  - Called by searchOptions to handle searching by vehicle
- vehicleHistSearch
  - called by vehicleHist to search for particular vehicles history