

CSCI 6511

Artificial Intelligence Project Part 3 – Project Report

Team 5

Sagar Sheth

Yuling Li

Akhil Dara

Table of Contents

1. Introduction.....	4
2. Templates & Facts	5
2.1 Station	5
2.2 NextStation	6
2.3 Line.....	7
2.4 Attraction.....	8
2.5 Fare.....	9
3. Relationship Between the templates.....	10
4. Additional Templates.....	11
4.1 Route	11
4.2 glbvisited	12
4.3 menuInfo.....	12
4.4 closedstations.....	13
5. Rules.....	14
5.1 menu.....	14
5.2 Station info.....	16
5.3 Fare info.....	17
5.4 Attraction info	18
5.5 Line Onfo	19
5.6 Closed Station.....	20
5.7 Print Closed Station.....	21
5.8 Reopen Station	22
5.9 Route info.....	23
5.10 Attraction Route.....	25
6. Running the Chatbot	27
6.1 Menu.....	27
6.2 Station Info	29
6.3 Fare Info	30
6.4 Attraction Info.....	32
6.5 Line Info.....	33
6.6 Closed Station.....	34
6.7 Print Closed Station.....	35
6.8 Reopen Station	36
6.9 Route Info	37
6.10 Attraction Info.....	39
6.11 Exit	41

1. Introduction

The project's goal is to create an expert system for the London Underground Metro System using CLIPS. The system will work on Zones 1 and 2 and will work on stations that exist on the following Lines;

- Bakerloo
- Central
- Circle
- District
- Hammersmith & City
- Jubilee
- Metropolitan
- Northern
- Piccadilly
- Victoria
- Waterloo & City

In this part of the project we have designed and developed initial set of templated and facts which would be the building blocks required to build the expert system. The facts were developed based on the data gathered from the following websites.

- <https://tfl.gov.uk/maps/track/tube>
- https://en.wikipedia.org/wiki/List_of_London_Underground_stations

The following are the initial list of templates designed to develop the expert system.

- Station
- NextStation
- Line
- Attraction
- Fare
- Route
- glbvisited
- menuInfo

2. Templates & Facts

2.1 Station

This template will capture the following information about each station.

- Name, name of the station.
- Line, the lines the station belongs to. Can have multiple values.
- Zone, the zone the station belongs to.
- Interchange, whether the station is an interchange station. Can have values either Yes/No.
- InternalInterchange, whether the station is an Internal Interchange station. Can have values either Yes/No.

Clips deftemplate

```
(deftemplate Station
  (slot name (type SYMBOL))
  (multislot line (type SYMBOL))
  (slot zone (type INTEGER))
  (slot Interchange (type SYMBOL) (allowed-symbols yes no) (default no))
  (slot InternalInterchange (type SYMBOL) (allowed-symbols yes no) (default no))
)
```

Clips deffacts Sample

```
(defacts Station
  (Station (name Aldgate)(line Metropolitan Circle)(zone 1)(Interchange Yes)(InternalInterchange No))
)
```

2.2 NextStation

This template will capture the following information about each station.

- CurrStation, name of the current station.
- CurrStationLine, the lines the current station belongs to. Can have multiple values.
- NextStation, name of the next station.
- NextStationLine, the lines the next station belongs to. Can have multiple values.

Clips deftemplate

```
(deftemplate NextStation
  (slot CurrStation (type SYMBOL))
  (multislot CurrStationLine (type SYMBOL))
  (slot NextStation (type SYMBOL))
  (multislot NextStationLine (type SYMBOL))
)
```

Clips deffacts Sample for Algate Station

```
(defacts NextStation
  (NextStation (CurrStation Aldgate)(CurrStationLine Metropolitan Circle)(NextStation TowerHill)(NextStationLine District Circle))
  (NextStation (CurrStation Aldgate)(CurrStationLine Metropolitan Circle)(NextStation LiverpoolStreet)(NextStationLine Metropolitan Central
  Circle HammersmithandCity))
)
```

2.3 Line

This template will capture the following information about each Line.

- name, name of the current Line.
- startingStation, the starting station of the Line from a specific direction. Note that this might not be the true starting Location, as we are only considering Zone 1 & 2. Additionally may change based on direction. May have multiple Values.
- endingStation, the ending station of the Line from a specific direction. Note that this might not be the true starting Location, as we are only considering Zone 1 & 2. Additionally may change based on direction. May have multiple Values.
- stationList, this documents the list of stations in this line. Can have multiple values.
- transferBetween, this an attribute which may be used in future parts to help identify the transfer stations on the line. Currently values are null.

Clips deftemplate

```
(deftemplate Line "Description of lines"
  (slot name (type SYMBOL))
  (multislot startingStation (type SYMBOL))
  (multislot endingStation (type SYMBOL))
  (multislot stationList (type SYMBOL))
  (multislot transferBetween (type SYMBOL) (default nil))
)
```

Clips deffacts Sample for WaterlooandCity Line

```
(deffacts Line
  (Line (name WaterlooandCity)(startingStation Waterloo)(endingStation Bank)(stationList Waterloo Bank)(transferBetween ))
)
```

2.4 Attraction

This template will capture the following information about each Attraction.

- name, name of the Attraction.
- nearbyStations, will state the nearby station for the Attraction. If An attraction can be reached by multiple stations, there will be a separate facts for the same.
- Description, gives a short description for the attracton.

Clips deftemplate

```
(deftemplate Attraction "the attractions in London area"  
  (slot name (type SYMBOL))  
  (multislot nearbyStations (type SYMBOL))  
  (slot description (type STRING))  
)
```

Clips deffacts Sample for RoyalAlbertHall

```
(deffacts Attraction  
  (Attraction (name BritishMuseum)(nearbyStations RussellSquare)(description "A vast treasure trove of global art and artifacts,  
including the Rosetta Stone."))  
)
```


2.5 Fare

This template will capture the following information about travel fare.

- StartStationZone, The Zone of the station the user starts the travel in.
- DestinationStationZone, The Zone of the station the user will end their trip.
- amount, the fare for travel between Zones. Has 4 enteries.

Clips deftemplate

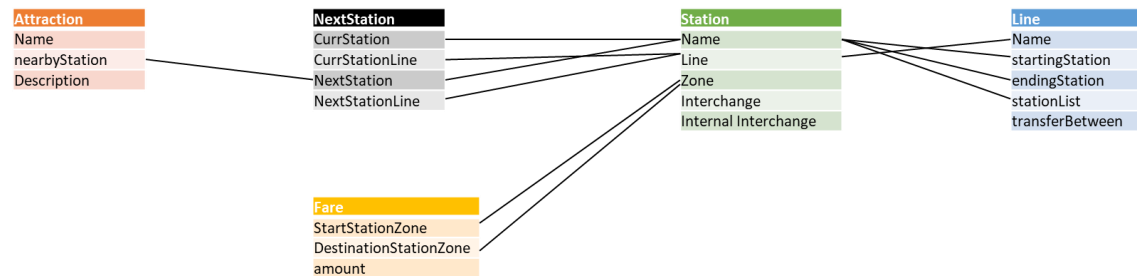
```
(deftemplate fare
  (slot StartStationZone (type INTEGER))
  (slot DestinationStationZone (type INTEGER))
  (slot amount (type FLOAT))
)
```

Clips deffacts

```
(deffacts fare
  (fare (StartStationZone 1)(DestinationStationZone 1)(amount 2.4))
  (fare (StartStationZone 2)(DestinationStationZone 2)(amount 2.4))
  (fare (StartStationZone 1)(DestinationStationZone 2)(amount 2.9))
  (fare (StartStationZone 2)(DestinationStationZone 1)(amount 2.9))
)
```

3. Relationship Between the templates

Below Image shows the relationship between the templates.



Kindly note If an attribute a is linked with an attribute b in another template, and that other attribute b is linked with another attribute c in a third template; Attribute a and c are directly linked but are not shown as part of the image to make the image more readable.

4. Additional Templates

4.1 Route

This template will capture the following information about the route.

- stations, will store the list of stations in sequential order of the path taken.
- currStn, temp variable which will store the current station our route is on.
- visited, will store the list of stations visited.
- count, will store the count of stations.

Clips deftemplate

```
(deftemplate Route
  (multislot stations (type SYMBOL))
  (slot currStn (type SYMBOL))
  (multislot visited (type SYMBOL))
  (slot count (type INTEGER))
)
```

Clips deffacts

As this is a fact created at run time there will be no preloaded facts into the system.

4.2 glbvisited

This template will capture the following information about the all the routes generated by the system.

- vst, will store the list of stations visited by all the routes generated by the system.

Clips deftemplate

```
(deftemplate glbvisited
  (multislot vst (type SYMBOL))
)
```

Clips deffacts

This is a fact created at run time, by default we will store start as the first Value.

4.3 menuInfo

This template will capture the following information.

- mInfo, will store the option the end user wants.

Clips deftemplate

```
(deftemplate menuInfo
  (slot mInfo (type STRING))
)
```

Clips deffacts

This is a fact that is modified at run time, by default we will store Main menu as the first Value.

```
(deffacts menuInfo
  (menuInfo( mInfo "Main Menu"))
)
```

4.4 closedstations

This template will store the following information for the system.

- cls, List of all Closed stations.

Clips deftemplate

```
(deftemplate closedstations
  (multislot cls (type SYMBOL))
)
```

Clips deffacts

This is a fact created at run time, by default we will store Closed as the first Value.

5. Rules

5.1 menu

This rule will generate the Menu required for the Underground Metro Chatbot. It will have the following options.

- Station Info: Information Regarding Station.
- Fare Info: Fare information.
- Attraction Info: Information Regarding Attractions.
- Line Info: Information regarding Lines.
- Closed Station: Report if a station has been closed.
- Print Closed Station: Get the List of Closed Stations.
- Reopen Station :Remove Station from Closed list.
- Route Info: Based on start and end station what route should be taken.
- Attraction Route: Based on a start station what route should be taken to reach the Attraction.

Users can input any of the following values to access the different parts of the menu.

- Station Info
- Fare info
- Attraction Info
- Line Info
- Closed Station
- Print Closed Station
- Reopen Station
- Route info
- Attraction Route

Clips defrule

```
;;*****  
;;  
;; menu.clp  
;;  
;; this file contains the rule which will generate the menu required for the London Underground Metro Chatbot  
;;*****  
(defrule menu  
  ?v <- (menuInfo (mInfo ?mInfo))  
  (test (eq ?mInfo "Main Menu"))  
  =>  
  (printout t crlf)  
  (printout t "Hello I am chatbot created to help you travel the London Underground Metro System" crlf)  
  (printout t "I am currently limited to stations within Zone 1 and 2" crlf)  
  
  (printout t "Kindly select how I can assist you today by selected one of the below options" crlf)  
  (printout t "Information" crlf)  
  (printout t "Station Info :Information Regarding Station." crlf)  
  (printout t "Fare Info :Fare information." crlf)  
  (printout t "Attraction Info :Information Regarding Attractions." crlf)  
  (printout t "Line Info :Information Regarding Lines." crlf)  
  (printout t "Closed Station :Report if a station has been closed." crlf)  
  (printout t "Print Closed Station :Get the List of Closed Stations." crlf)  
  (printout t "Reopen Station :Remove Station from Closed list." crlf)  
  (printout t crlf)  
  
  (printout t "Point to Point Travel Information" crlf)  
  (printout t "Route Info :Based on start and end station what route should be taken" crlf)  
  (printout t "Attraction Route :Based on a start station what route should be taken to reach the Attraction" crlf)  
  
  (printout t crlf)  
  
  (printout t "Exit: Enter Exit to close this chatbot" crlf)  
  (modify ?v (mInfo (readline)))  
  (printout t crlf)  
)
```

5.2 Station info

To retrieve information about the Station we have 5 rules defined

- Info-Station: Will retrieve station name from end user.
- Check-Info-Station: Will check if the input values exists as a Station.
- get-station-info-response: Will print out the following information about the station,
 - Station Name
 - Zone
 - Lines the station Exists on
 - Interchange
 - Internal Interchange
- get-station-info-response2: Will print all the next stations of the current station
- get-menu-back: Will return the program back to the main menu and retract the required temporary facts.

Clips defrule

```
(defrule Info-Station
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Station Info"))
  =>
  (printout t "Enter Station" crlf)
  (assert (Stn (read)))
)
(defrule Check-Info-Station
  ?st <- (Stn ?Stn)
  ?mi <- (menuInfo (mInfo ?mInfo))
  (not (Station (name ?Stn)))
  =>
  (printout t "Station with name "" ?Stn "" does not exist." crlf)
  (retract ?st)
  (modify ?mi (mInfo "Main Menu"))
)
(defrule get-station-info-response
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Station Info"))
  ?st <- (Stn ?Stn)
  (Station (name ?Stn)(line $?lines)(zone ?zone)(Interchange ?Interchange)(InternalInterchange ?InternalInterchange))
  =>
  (printout t "Station Name:" ?Stn crlf)
  (printout t "Zone:" ?zone crlf)
  (foreach ?line $?lines
    (printout t "Station Line:" ?line crlf))
  (printout t "Interchange:" ?Interchange crlf)
  (printout t "Internal Interchange:" ?InternalInterchange crlf)
  (modify ?mi (mInfo "Station Info2"))
)
(defrule get-station-info-response2
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Station Info2"))
  ?st <- (Stn ?Stn)
  (NextStation (CurrStation ?Stn)(NextStation ?NextStation))
  =>
  (printout t "Next Station:" ?NextStation crlf)
)
(defrule get-menu-back
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?st <- (Stn ?Stn)
  (test (eq ?mInfo "Station Info2"))
  =>
  (retract ?st)
  (modify ?mi (mInfo "Main Menu"))
)
```


5.3 Fare info

To retrieve information about the Station we have 4 rules defined

- Info-Fare: Will retrieve the Start and End Station Name from the end User
- Check-Info-Fare-st: Will check if the start station provided by the user exists.
- Check-Info-Fare-end: Will check if the start station provided by the user exists.
- get-fare-info-response: Will print out the following information about the station, retract the temporary facts and return the program to the main menu.
 - Start Station Zone
 - End Station Zone
 - Fare

Clips defrule

```
(defrule Info-Fare
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Fare Info"))
  =>
  (printout t "Enter Start Station" crlf)
  (assert (stStn (read)))
    (printout t "Enter End Station" crlf)
  (assert (endStn (read)))
)
(defrule Check-Info-Fare-st
  ?st <- (stStn ?stStn)
    ?mi <- (menuInfo (mInfo ?mInfo))
    (not (Station (name ?stStn)))
  =>
  (printout t "Station with name " ?stStn " does not exist." crlf)
  (retract ?st)
    (modify ?mi (mInfo "Main Menu"))
)
(defrule Check-Info-Fare-end
  ?end <- (endStn ?endStn)
    ?mi <- (menuInfo (mInfo ?mInfo))
    (not (Station (name ?endStn)))
  =>
  (printout t "Station with name " ?endStn " does not exist." crlf)
  (retract ?end)
    (modify ?mi (mInfo "Main Menu"))
)
(defrule get-fare-info-response
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Fare Info"))
  (Attraction (name ?name) (nearbyStations ?nearbyStations) (description ?description))
  ?st <- (stStn ?stStn)
    ?end <- (endStn ?endStn)
    (Station (name ?stStn) (zone ?startZone))
    (Station (name ?endStn) (zone ?endZone))
    (fare (StartStationZone ?startZone) (DestinationStationZone ?endZone) (amount ?amt))
  =>
  (printout t "Zone of " ?stStn ": " ?startZone crlf)
  (printout t "Zone of " ?endStn ": " ?endZone crlf)
  (printout t "Fare is:" ?amt crlf)

  (retract ?st)
  (retract ?end)
  (modify ?mi (mInfo "Main Menu"))
)
```

5.4 Attraction info

To retrieve information about the Station we have 3 rules defined

- Info-Attraction: Will retrieve name of the Attraction from the end user.
- Check-Attraction-Info: Will check if the information provided as input exists.
- get-attraction-info-response: Will print out the following information about the station, retract the temporary facts and return the program to the main menu.
 - Attraction Name
 - Description
 - Nearby Station/s

Clips defrule

```
(defrule Info-Attraction
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Attraction Info"))
  =>
  (printout t "Enter Attraction name, you want information about." crlf)
  (assert (attrName (read))))
)

(defrule Check-Attraction-Info
  ?an <- (attrName ?attrName)
  ?mi <- (menuInfo (mInfo ?mInfo))
  (not (Attraction (name ?attrName)))
  =>
  (printout t "Attraction with name "" ?attrName "" does not exist." crlf)
  (retract ?an)
  (modify ?mi (mInfo "Main Menu"))
)

(defrule get-attraction-info-response
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Attraction Info"))
  (Attraction (name ?name) (nearbyStations $?nearbyStations) (description ?description))
  ?an <- (attrName ?attrName)
  (test (eq $?name $?attrName))
  =>
  (printout t "Attraction " ?name crlf)
  (printout t ?description crlf)
  (printout t "Neaby Station:" $?nearbyStations crlf)

  (retract ?an)
  (modify ?mi (mInfo "Main Menu"))
)
```

5.5 Line Onfo

To retrieve information about the Line we have 3 rules defined

- Info-Line: Will retrieve the Line Name from the end User
- Check-Line-Info: Will check if the Line provided by the user exists.
- get-line-info-response: Will print out the following information about the Line, retract the temporary facts and return the program to the main menu.
 - Line Name
 - List of stations within the line

Clips defrule

```
(defrule Info-Line
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Line Info"))
  =>
  (printout t "Enter Line name, you want information about." crlf)
  (assert (lineName (read))))
)

(defrule Check-Line-Info
  ?ln <- (lineName ?lineName)
  ?mi <- (menuInfo (mInfo ?mInfo))
  (not (Line (name ?lineName)))
  =>
  (printout t "Line with name "" ?lineName "" does not exist." crlf)
  (retract ?ln)
  (modify ?mi (mInfo "Main Menu"))
)

(defrule get-line-info-response
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Line Info"))
  (Line (name ?lineName) (stationList $?stationList))
  ?ln <- (lineName ?lineName)
  =>
  (printout t "Line Name: " ?lineName crlf)
  (printout t "Stations within the line: " crlf)
  (foreach ?stlst $?stationList
    (printout t ?stlst crlf))
  (printout t crlf)

  (retract ?ln)
  (modify ?mi (mInfo "Main Menu"))
)
```

5.6 Closed Station

To report that a station has been closed we have 3 rules defined

- Info-Closed-Stations: Will retrieve the Station Name from the end User
- Check-Closed-Stations: Will check if the Station provided by the user exists.
- Store-Closed-Stations: Will store the information provided by the user in closedstations fact.

Clips defrule

```
(defrule Info-Closed-Stations
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Closed Station")))
=>
  (printout t "Enter Station name, you want report as closed." crlf)
  (assert (clsStn (read)))
)

(defrule Check-Closed-Stations
  ?cl <- (clsStn ?clsStn)
  ?mi <- (menuInfo (mInfo ?mInfo))
  (not (Station (name ?clsStn)))
=>
  (printout t "Station with name " ?clsStn " does not exist." crlf)
  (retract ?cl)
  (modify ?mi (mInfo "Main Menu"))
)

(defrule Store-Closed-Stations
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Closed Station"))
  ?cl <- (clsStn ?clsStn)
  ?clst<-(closedstations (cls $?cls))
=>
  (modify ?clst (cls $?cls ?clsStn))
  (printout t "Station " ?clsStn " added to closed station List" crlf)
  (retract ?cl)
  (modify ?mi (mInfo "Main Menu"))
)
```

5.7 Print Closed Station

This rule will print all existing closed stations defined in the system.

Clips defrule

```
(defrule Print-Closed-Station
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Print Closed Station"))
  ?clst<-(closedstations (cls $?cls))

  =>
    (foreach ?cl $?cls
      (printout t ?cl crlf))
    (modify ?mi (mInfo "Main Menu"))
)
```

5.8 Reopen Station

To report that a station has been reopened we have 3 rules defined

- Info-Reopen-Station: Will retrieve the Station Name from the end User
- Check-Reopen-Station: Will check if the Station provided by the user exists.
- Final-Reopen-Station: Will delete the Station provided by the user from the closedstations fact.

Clips defrule

```
(defrule Info-Reopen-Station
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Reopen Station"))
  =>
    (printout t "Enter Station name to check:" crlf)
    (assert (ropnstn (read)))
)

(defrule Check-Reopen-Station
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?rop <- (ropnstn ?ropnstn)
  ?x<-(closedstations (cls $?clsList))
  (test (not (member$ ?ropnstn $?clsList)))
  ;(not (member$ ?CurrStation $?visited)
  =>
    (printout t "Station with name "" ?ropnstn "" does not exist in closed station List." crlf)
    (retract ?rop)
    (modify ?mi (mInfo "Main Menu"))
)

(defrule Final-Reopen-Station
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?rop <- (ropnstn ?ropnstn)
  ?clst<-(closedstations (cls $?cls))
  (test (eq ?mInfo "Reopen Station"))
  =>
    (bind ?newClsList (delete-member$ ?cls ?ropnstn))
    (modify ?clst (cls ?newClsList))
    (printout t "Station with name "" ?ropnstn "" has been ReOpened." crlf)
    (retract ?rop)
    (modify ?mi (mInfo "Main Menu"))
)
```

5.9 Route info

To retrieve information about the Station we have 6 rules defined

- Info-Route: Will retrieve the Start and End Stations from the end user.
- start-dfs: Will initialize the route fact with the start station.
- dfs: Will build the best possible route between the start and end station.
- delete-route: Will delete unwanted routes from routes facts.
- print-route: Will print out the following information about the route, Additionally will retract the temporary facts and reset the glbvisited fact.
 - Route taken sequentially.
 - No of stations.
 - Fare.
- get-menu-back2: Will return the user to the main menu and clean all the route facts.

Clips defrule

```

(defrule Info-Route
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Route Info"))
  =>
  (printout t "Enter Start Station" crlf)
  (assert (stStn (read)))
    (printout t "Enter End Station" crlf)
  (assert (endStn (read)))
)

(defrule start-dfs
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Route Info"))
  ?st <- (stStn ?stStn)
  ?end <- (endStn ?endStn)
  (NextStation (CurrStation ?stStn) (NextStation ?next))
  =>
  (printout t "Route between:" ?stStn " and " ?endStn crlf)
  (assert (Route (stations ?stStn) (currStn ?stStn)(count 1)))
)

(defrule dfs
  ?route <- (Route (stations $?stations) (currStn ?currStn) (visited $?visited) (count ?count))
  ?end<- (endStn ?endStn)
  ?vt<-(glbvisited (vst $?vst))
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?clst<-(closedstations (cls $?cls))
  (test (eq ?mInfo "Route Info"))
  (NextStation (CurrStation ?CurrStation) (NextStation ?endStn &:(not (member$ ?CurrStation $?visited))))
  (NextStation (CurrStation ?currStn) (NextStation ?next &:(not (member$ ?next $?visited))))
  (test (not (member$ ?next $?vst)))
  (test (not (member$ ?next $?cls)))
  (not (eq ?currStn ?endStn))
  =>
  (assert (Route (stations $?stations ?next) (currStn ?next) (visited $?visited ?currStn) (count (+ ?count 1))))
  (modify ?vt (vst $?vst ?next))
)

(defrule delete-route
  ?end <- (endStn ?endStn)
  ?route <- (Route (currStn ?v &:(neq ?v ?endStn)))
  =>
  (retract ?route)
)

(defrule print-route
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?allroute <- (Route (stations $?stations) (currStn ?currStn) (count ?count))
  ?st <- (stStn ?stStn)
  ?end<- (endStn ?endStn)
  ?vt<-(glbvisited (vst $?vst))

  (Station (name ?stStn) (zone ?startZone))
  (Station (name ?endStn) (zone ?endZone))
  (fare (StartStationZone ?startZone) (DestinationStationZone ?endZone) (amount ?amt))
  (test (eq ?currStn ?endStn))
  (test (eq ?mInfo "Route Info"))
  ;
  =>
  (foreach ?station $?stations
    (printout t "->" ?station ))
    (printout t crlf)
    (printout t "No of Stations: " ?count crlf)
    (printout t "Fare: " ?amt crlf)
    (retract ?st)
    (retract ?end)
    (retract ?allroute)
    (modify ?vt (vst Start))
  )

(defrule get-menu-back2
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Route Info"))
  =>
  (do-for-all-facts ((?f Route)) (retract ?f))
  (modify ?mi (mInfo "Main Menu"))
)

```


5.10 Attraction Route

To retrieve information about the Station we have 6 rules defined

- Info-Attraction-Route: Will retrieve the Start Station and the destination Attraction from the end user.
- get-attraction-route-info-response: Will initialize the route fact with the start station.
- dfs-attr-route: Will build the best possible route between the start and end station.
- delete-route-attr-route: Will delete unwanted routes from routes facts.
- print-route: Will print out the following information about the route, Additionally will retract the temporary facts and reset the glbvisited fact.
 - Route taken sequentially.
 - No of stations.
 - Fare.
- get-menu-back3: Will return the user to the main menu and clean all the route facts.

Clips defrule

```

(defrule Info-Attraction-Route
  (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Attraction Route"))
  =>
  (printout t "Enter Attraction name, you want travel to" crlf)
  (assert (attrName (read)))
    (printout t "Enter Station you will start the journey from" crlf)
    (assert (stStn (read)))
  )

(defrule get-attraction-route-info-response
  ?mi <- (menuInfo (mInfo ?mInfo))
  (test (eq ?mInfo "Attraction Route"))
  ?an <- (attrName ?attrName)
  ?st <- (stStn ?stStn)
  (Attraction (name ?attrName) (nearbyStations ?nearbyStations) (description ?description))
  =>
    (printout t "Route between:" ?stStn " and " ?attrName ", through " ?nearbyStations crlf)
    (assert (endStn ?nearbyStations))
    (retract ?an)
    (assert (Route (stations ?stStn) (currStn ?stStn)(count 1)))
  )

(defrule dfs-attr-route
  ?route <- (Route (stations $?stations) (currStn ?currStn) (visited $?visited) (count ?count))
  ?end<- (endStn ?endStn)
  ?vt<-(glbvisited (vst $?vst))
  ?clst<-(closedstations (cls $?cls))
  (NextStation (CurrStation ?CurrStation) (NextStation ?endStn &:(not (member$ ?CurrStation $?visited))))
  (NextStation (CurrStation ?currStn) (NextStation ?next &:(not (member$ ?next $?visited))))
  (test (not (member$ ?next $?vst)))
  (test (not (member$ ?next $?cls)))
  (not (eq ?currStn ?endStn))
  =>
    (assert (Route (stations $?stations ?next) (currStn ?next) (visited $?visited ?currStn) (count (+ ?count 1))))
  (modify ?vt (vst $?vst ?next))
  )

(defrule delete-route-attr-route
  ?end <- (endStn ?endStn)
  ?route <- (Route (currStn ?v &:(neq ?v ?endStn)))
  =>
    (retract ?route)
  )

(defrule print-route
  ?mi <- (menuInfo (mInfo ?mInfo))
  ?allroute <- (Route (stations $?stations) (currStn ?currStn) (count ?count))
  ?st <- (stStn ?stStn)
  ?end<- (endStn ?endStn)
  ?vt<-(glbvisited (vst $?vst))

  (Station (name ?stStn) (zone ?startZone))
  (Station (name ?endStn) (zone ?endZone))
  (fare (StartStationZone ?startZone) (DestinationStationZone ?endZone) (amount ?amt))
  (test (eq ?currStn ?endStn))
  ; (test (eq ?mInfo "Route Info"))
  =>
    (foreach ?station $?stations
      (printout t "->" ?station ))
      (printout t crlf)
      (printout t "No of Stations: " ?count crlf)
      (printout t "Fare: " ?amt crlf)
      (retract ?st)
      (retract ?end)
      (retract ?allroute)
      (modify ?vt (vst Start))
    )

  (defrule get-menu-back3
    ?mi <- (menuInfo (mInfo ?mInfo))
    (test (eq ?mInfo "Attraction Route"))
    =>

      (do-for-all-facts ((?f Route)) (retract ?f))
      (modify ?mi (mInfo "Main Menu"))
  )

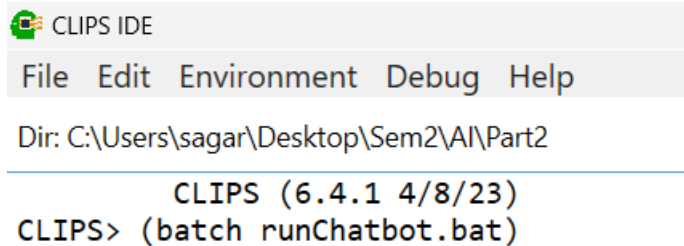
```

6. Running the Chatbot

6.1 Menu

Using the runChatbot.bat file we will load all the rules, templates and facts into clips.

```
(clear)
(load TravelerTemplates.clp)
(load TravelerFacts.clp)
(load menu.clp)
(load Rule.clp)
(reset)
(run)
```



CLIPS IDE

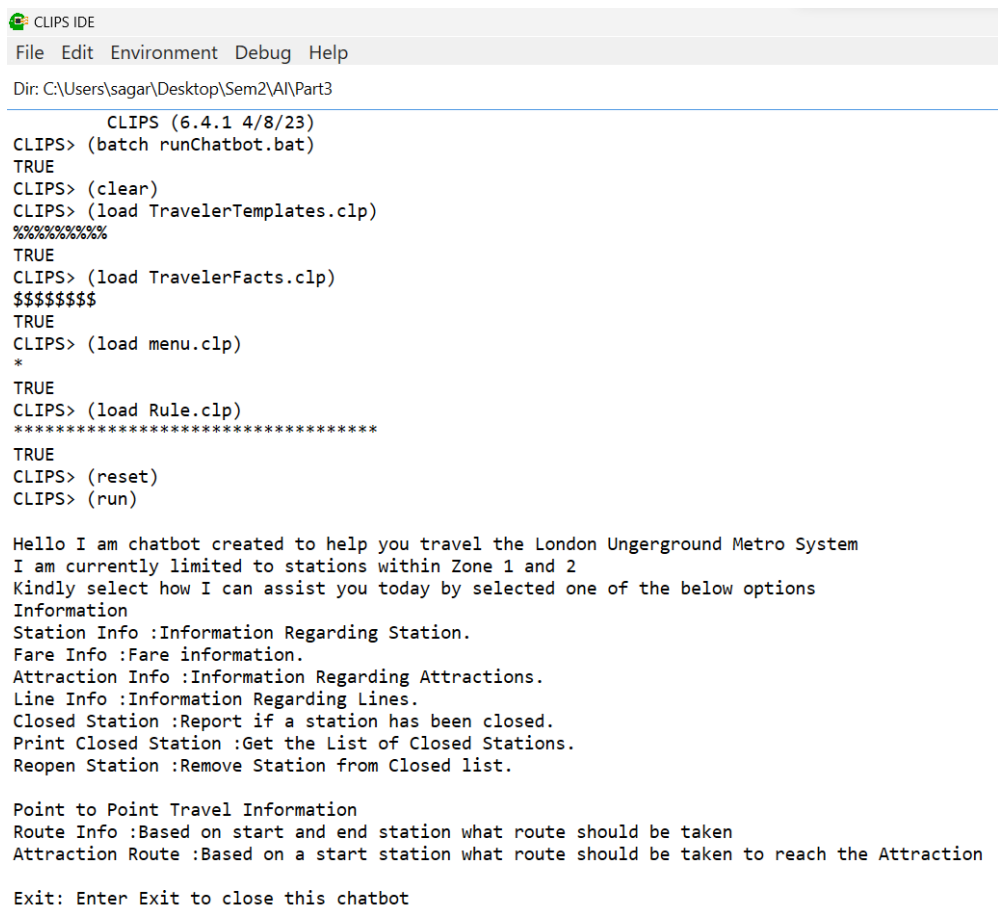
File Edit Environment Debug Help

Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2

CLIPS (6.4.1 4/8/23)

CLIPS> (batch runChatbot.bat)

Clips Output



CLIPS IDE

File Edit Environment Debug Help

Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3

CLIPS (6.4.1 4/8/23)

CLIPS> (batch runChatbot.bat)

TRUE

CLIPS> (clear)

CLIPS> (load TravelerTemplates.clp)

%%%%%%%%%

TRUE

CLIPS> (load TravelerFacts.clp)

\$\$\$\$\$\$\$

TRUE

CLIPS> (load menu.clp)

*

TRUE

CLIPS> (load Rule.clp)

TRUE

CLIPS> (reset)

CLIPS> (run)

Hello I am chatbot created to help you travel the London Ungerground Metro System

I am currently limited to stations within Zone 1 and 2

Kindly select how I can assist you today by selected one of the below options

Information

Station Info :Information Regarding Station.

Fare Info :Fare information.

Attraction Info :Information Regarding Attractions.

Line Info :Information Regarding Lines.

Closed Station :Report if a station has been closed.

Print Closed Station :Get the List of Closed Stations.

Reopen Station :Remove Station from Closed list.

Point to Point Travel Information

Route Info :Based on start and end station what route should be taken

Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot

If the User enters an unknown Menu option that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Hello I am chatbot created to help you travel the London Ungerground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Hello

Inccorrect Menu option

Hello I am chatbot created to help you travel the London Ungerground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
```

6.2 Station Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2
2. Fare Info :Fare information.
3. Attraction Info :Information Regarding Attractions.

Point to Point Travel Information
4. Route Info :Based on start and end station what route should be taken
5. Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Type anything execept the above options to exit this chatbot
Station Info

Enter Station
Bank
Station Name:Bank
Zone:1
Station Line:WaterlooandCity
Station Line:northern
Station Line:Central
Interchange:yes
Internal Interchange:yes
Next Station:Moorgate
Next Station:LondonBridge
Next Station:Waterloo
Next Station:LiverpoolStreet
Next Station:StPauls
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
kindly select how i can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Station Info

Enter Station
Unknown
Station with name 'Unknown' does not exist.
```

6.3 Fare Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2
next Station: StPauls

Hello I am chatbot created to help you travel the London Ungerground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
1. Station Info :Information Regarding Station.
2. Fare Info :Fare information.
3. Attraction Info :Information Regarding Attractions.

Point to Point Travel Information
4. Route Info :Based on start and end station what route should be taken
5. Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Type anything execept the above options to exit this chatbot
Fare Info

Enter Start Station
Aldgate
Enter End Station
AldgateEast
Zone of Aldgate: 1
Zone of AldgateEast: 1
Fare is:2.4
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Fare Info

Enter Start Station
Unknown
Enter End Station
Bank
Station with name 'Unknown' does not exist.
```

 CLIPS IDE

File Edit Environment Debug Help

Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3

Station Info :Information Regarding Station.

Fare Info :Fare information.

Attraction Info :Information Regarding Attractions.

Line Info :Information Regarding Lines.

Closed Station :Report if a station has been closed.

Print Closed Station :Get the List of Closed Stations.

Reopen Station :Remove Station from Closed list.

Point to Point Travel Information

Route Info :Based on start and end station what route should be taken

Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot

Fare Info

Enter Start Station

Bank

Enter End Station

Unknown

Station with name 'Unknown' does not exist.

6.4 Attraction Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2
Zone of Aldgate: 1
Zone of AldgateEast: 1
Fare is:2.4

Hello I am chatbot created to help you travel the London Ungerground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
1. Station Info :Information Regarding Station.
2. Fare Info :Fare information.
3. Attraction Info :Information Regarding Attractions.

Point to Point Travel Information
4. Route Info :Based on start and end station what route should be taken
5. Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Type anything execept the above options to exit this chatbot
Attraction Info

Enter Attraction anme, you want information about.
TateBritain
Attraction TateBritain
Art museum with a vast collection of British artworks.
Neaby Station:Pimlico
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Attraction Info

Enter Attraction name, you want information about.
Unknown
Attraction with name 'Unknown' does not exist.
```


6.5 Line Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Line Info

Enter Line name, you want information about.
Bakerloo
Line Name: Bakerloo
Stations within the line:
KensalGreen
QueensPark
KilburnPark
MaidaVale
WarwickAvenue
Paddington
EdgwareRoad
Marylebone
BakerStreet
RegentsPark
OxfordCircus
PiccadillyCircus
CharingCross
Embankment
Waterloo
LambethNorth
ElephantandCastle
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Line Info

Enter Line name, you want information about.
Unknown
Line with name 'Unknown' does not exist.
```

6.6 Closed Station

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
kindly select how i can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Closed Station

Enter Station name, you want report as closed.
Bank
Station Bank added to closed station List
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
kindly select how i can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Closed Station

Enter Station name, you want report as closed.
Unknown
Station with name 'Unknown' does not exist.
```

6.7 Print Closed Station

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
I am currently limited to stations within zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Print Closed Station

Closed
Bank
```

6.8 Reopen Station

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
kindly select how i can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Reopen Station

Enter Station name to check:
Bank
Station with name 'Bank' has been ReOpened.
```

If the User enters a value that does not exists, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
kindly select how i can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Reopen Station

Enter Station name to check:
Bank
Station with name 'Bank' does not exist in closed station List.
```

6.9 Route Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2

Hello I am chatbot created to help you travel the London Underground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
1. Station Info :Information Regarding Station.
2. Fare Info :Fare information.
3. Attraction Info :Information Regarding Attractions.

Point to Point Travel Information
4. Route Info :Based on start and end station what route should be taken
5. Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Type anything execept the above options to exit this chatbot
Route Info

Enter Start Station
Aldgate
Enter End Station
AldgateEast
Route between:Aldgate and AldgateEast
->Aldgate->LiverpoolStreet->AldgateEast
No of Stations: 3
Fare: 2.4
```

If you close a station an alternate route will be shown

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3

Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Route Info

Enter Start Station
Aldgate
Enter End Station
AldgateEast
Route between:Aldgate and AldgateEast
->Aldgate->TowerHill->AldgateEast
No of Stations: 3
Fare: 2.4
```

If the User enters a value that does not exist, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help

Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Route Info

Enter Start Station
Unknown
Enter End Station
Bank
Station with name 'Unknown' does not exist.
```

```
CLIPS IDE
File Edit Environment Debug Help

Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Route Info

Enter Start Station
Bank
Enter End Station
Unknown
Station with name 'Unknown' does not exist.
```

6.10 Attraction Info

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part2

Hello I am chatbot created to help you travel the London Underground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
1. Station Info :Information Regarding Station.
2. Fare Info :Fare information.
3. Attraction Info :Information Regarding Attractions.

Point to Point Travel Information
4. Route Info :Based on start and end station what route should be taken
5. Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Type anything execept the above options to exit this chatbot
Attraction Route

Enter Attraction name, you want travel to
BritishMuseum
Enter Station you will start the journey from
Bank
Route between:Bank and BritishMuseum, through RussellSquare
->Bank->StPauls->ChanceryLane->Holborn->RussellSquare
No of Stations: 5
Fare: 2.4
```

If A station is closed an alternate route will be shown.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3

Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Attraction Route

Enter Attraction name, you want travel to
BritishMuseum
Enter Station you will start the journey from
Bank
Route between:Bank and BritishMuseum, through RussellSquare
->Bank->Moorgate->Barbican->Farringdon->KingsCrossStPancras->RussellSquare
No of Stations: 6
Fare: 2.4
```

If the User enters a value that does not exist, then the System will print a message and return to the menu.

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Attraction Route

Enter Attraction name, you want travel to
Unknown
Enter Station you will start the journey from
Bank
Attraction with name 'Unknown' does not exist.
```

```
CLIPS IDE
File Edit Environment Debug Help
Dir: C:\Users\sagar\Desktop\Sem2\AI\Part3
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Attraction Route

Enter Attraction name, you want travel to
BritishMuseum
Enter Station you will start the journey from
Unknown
Station with name 'Unknown' does not exist.
```


6.11 Exit

Hello I am chatbot created to help you travel the London Underground Metro System
I am currently limited to stations within Zone 1 and 2
Kindly select how I can assist you today by selected one of the below options
Information
Station Info :Information Regarding Station.
Fare Info :Fare information.
Attraction Info :Information Regarding Attractions.
Line Info :Information Regarding Lines.
Closed Station :Report if a station has been closed.
Print Closed Station :Get the List of Closed Stations.
Reopen Station :Remove Station from Closed list.

Point to Point Travel Information
Route Info :Based on start and end station what route should be taken
Attraction Route :Based on a start station what route should be taken to reach the Attraction

Exit: Enter Exit to close this chatbot
Exit

Thank you for using this chat bot.
CLIPS>