

Access to bus data is provided through a dictionary called `shared_data`.

There are three keys to `shared_data`: `shared_data['locations']`, `shared_data['routes']`, and `shared_data['stops']`. These hold data on bus locations, bus routes, and bus stops, respectively. The JSON formats are defined below.

```
# for each bus
[
    float "GroundSpeed" # mph
    int "Heading" # degrees
    bool "IsDelayed"
    bool "IsOnRoute"
    float "Latitude"
    float "Longitude"
    string "Name"
    int "RouteId"
    int "Seconds"
    string "TimeStamp" # javascript date
    int "VehicleID"
]
```

```
# for each route
[
    string "Description" # e.g. "Blue", "Red", etc.
    int "ETATypeID"
    string "EncodedPolyline" # no idea how this works
    bool "HideRouteLine"
    string "InfoText" # usually empty
    bool "IsCheckLineOnlyOnMap"
    bool "IsCheckedOnMap"
    bool "IsVisibleOnMap"
    "Landmarks": [] # usually empty
    float "MapLatitude"
    string "MapLineColor" # html hex color, e.g. "#0000FF"
    float "MapLongitude"
    int "MapZoom"
    int "Order"
    int "RouteID"
    bool "ShowPolygon"
    bool "ShowRouteArrows"
    string "StopTimesPDFLink" # all of these links are broken
    string "TextingKey" # usually empty
    bool "UseScheduleTripsInPassengerCounter"
    "Stops": [
        # for each stop
        int "AddressID"
        string "City" # not always accurate
        float "Latitude"
```

```

    string "Line1"
    string "Line2" # usually empty
    float "Longitude"
    string "State" # always equal to "CT"
    string "Zip"
    string "Description"
    int "Heading"
    "MapPoints": [] # usually empty
    int "MaxZoomLevel"
    int "Order"
    string "RouteDescription" # usually empty
    int "RouteID"
    int "RouteStopID"
    int "SecondsAtStop"
    int "SecondsToNextStop"
    bool "ShowDefaultedOnMap"
    bool "ShowEstimatesOnMap"
    string "SignVerbiage"
    string "TextingKey"
  ]
]

```

```

# for each stop
[
  int "RouteID"
  int "RouteStopID"
  "ScheduledTimes": [
    # list has a maximum length of two
    {
      string "ArrivalTimeUTC" # javascript date
      int "AssignedVehicleId"
      string "DepartureTimeUTC" # javascript date
    }
  ]
  "VehicleEstimates": [
    # likewise this list also has maximum length of two
    {
      bool "OnRoute"
      int "SecondsToStop"
      int "VehicleID"
    }
  ]
]

```

If you want to look at some example data, here are links to the [locations](#), [routes](#), and [stops](#). Note that if you visit these pages while there are no buses running you will not get any data.