

## Data Scanner For .NET

### .NET Framework Version: 2.0

*Data Scanner* is an application developed for the Android platform. However, it provides users with the ability to export scan information to a text file. From the text file, it is then possible to download the data to a Windows computer. Downloading and processing the data in Windows is the goal of .NET development for *Data Scanner*. The *DataScanner* class provides developers with multiple methods and properties for processing exported scans from *Data Scanner*.

**Note:** In order to download the export file from the device for processing on Windows, the device must first be mounted. For instructions on mounting see your device documentation. An unmounted device will throw a *ScannerException* when attempting to access the scanner.

### **Public Class *DataCollectionScanner*. *DataScannerDownloader***

This is the primary class for processing scans. Below is a list and descriptive summary of each public property and method provided by *DataScanner*.

#### *Public Enum*

*LocationType*: Used to return the type of location stored with the scan. The possible values are: *LocationType.Coordinates* and *LocationType.PhysicalAddress*.

#### *Public Properties*

Version 1.0 of *DataScanner* does not have any public properties.

#### *Public Methods*

Method Name	Return Type	Description
ClearScanner	None	Deletes the saved export file from the Android device.
DeviceAvailable	Boolean	Returns True if the Android device is attached and the export file exists. Otherwise, returns False.
DownloadScanner	Array of ScanRecord items	Retrieves all scans stored in export file and parses them into an array of ScanRecord objects.
ScanCount	Integer	Returns a count of the total number of scans stored in the export file.
ScansAvailable	Boolean	Returns True if there are scans in the export file. Otherwise, False.

**Public Class DataCollectionScanner.ScannerException**

This class defines a custom Exception. This exception is thrown by the DownloadScanner method of the DataScanner class when the export file cannot be found on the device. Generally, this will be a result of not having completed an export operation inside the DataScanner application prior to attempting to download scans.

**DataCollectionScanner.DataScannerDownloader.ScanRecord Class**

The ScanRecord class parses the information in a given scan record and returns it to the developer in an easily processed form. Note that, for consistency, the ScanRecord class uses the same terminology as the Android LBS (Location Based Services) Location class.

*Public ReadOnly Properties*

Property Name	Return Type	Description
AdminArea	String	The AdminArea of the ScanRecord. In the US this corresponds to the State. Note that this will be an empty string if GetScanRecordLocationType returns LocationType.Coordinates.
BarCode	String	This is the actual barcode associated with the ScanRecord
CountryName	String	The country of the ScanRecord. Note that this will be an empty string if GetScanRecordLocationType returns LocationType.Coordinates
Latitude	Double	The latitude of the ScanRecord. Note that this will return 1000 if GetScanRecordLocationType returns LocationType.PhysicalAddress
Locality	String	The Locality of the ScanRecord. This corresponds to the City/Town. Note that this will be an empty string if GetScanRecordLocationType returns LocationType.Coordinates.
LocationCoordinate	PointF	Returns a PointF containing the Longitude and Latitude of the

		ScanRecord. X=Longitude, Y=Latitude
Longitude	Double	The longitude of the ScanRecord. Note that this will return 1000 if GetScanRecordLocationType returns LocationType.PhysicalAddress
PostalCode	String	Returns the Postal/Zip Code of the ScanRecord. Note that this will be an empty string if GetScanRecordLocationType returns LocationType.Coordinates
ScanDate	Date	Returns the Date (and Time) the barcode was scanned
Thoroughfare	String	The Thoroughfare of the ScanRecord. This corresponds to the street name (and sometimes building number) for the ScanRecord. Note that this will be an empty string if GetScanRecordLocationType returns LocationType.Coordinates.

*Public Methods*

<b>Method Name</b>	<b>Return Type</b>	<b>Description</b>
GetScanRecordLocationType	LocationType (Enum)	Returns the LocationType for the ScanRecord object.
ToString	String	Returns a string containing all information for the ScanRecord. The value of ToString will be different for LocationType.PhysicalAddress than it will be for LocationType.Coordinates.