|  |
| --- |
| [Type the company name] |
| EmvLib Reference |
| [Type the document subtitle] |

|  |
| --- |
| [Type the author name]  [Pick the date] |

Table of Contents

[Press ALT+A and F9 to update the TOC and other fields]

EmvLib Reference

Namespaces

[EmvLib](#topic_0000000000000001)

EmvLib Namespace

Classes

[AflEntry](#topic_0000000000000082), [AflResult](#topic_0000000000000080), [CAKeyElement](#topic_0000000000000002), [CAKeys](#topic_0000000000000009), [CaKey](#topic_000000000000000D), [CaKeyStore](#topic_000000000000000B), [CardNotSupportedExceprion](#topic_0000000000000036), [EmvConstants](#topic_0000000000000014), [OfflineAuth](#topic_000000000000003D), [SCardReaderExtensions](#topic_0000000000000047), [SmartApplication](#topic_0000000000000049), [SmartCard](#topic_0000000000000059), [SmartEmvRecord](#topic_0000000000000062), [SmartTag](#topic_000000000000006E), [SmartTlv](#topic_0000000000000069), [StringTools](#topic_000000000000007B), [TlvTools](#topic_0000000000000087)

Enumerations

[CertificateType](#topic_000000000000003A)

AflEntry Class

|  |  |
| --- | --- |
| C# |  |
| public class AflEntry | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Fields

[EndRecord](#topic_0000000000000085), [OfflineRecords](#topic_0000000000000086), [Sfi](#topic_0000000000000083), [StartRecord](#topic_0000000000000084)

EndRecord Field

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) EndRecord | |

Source code

|  |
| --- |
| public int EndRecord; |

See Also

Applies to: [AflEntry](#topic_0000000000000082)

OfflineRecords Field

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) OfflineRecords | |

Source code

|  |
| --- |
| public int OfflineRecords; |

See Also

Applies to: [AflEntry](#topic_0000000000000082)

Sfi Field

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) Sfi | |

Source code

|  |
| --- |
| public int Sfi; |

See Also

Applies to: [AflEntry](#topic_0000000000000082)

StartRecord Field

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) StartRecord | |

Source code

|  |
| --- |
| public int StartRecord; |

See Also

Applies to: [AflEntry](#topic_0000000000000082)

AflResult Class

|  |  |
| --- | --- |
| C# |  |
| public class AflResult | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Fields

[AflEntries](#topic_0000000000000081)

AflEntries Field

|  |  |
| --- | --- |
| C# |  |
| new public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[AflEntry](#topic_0000000000000082)> AflEntries | |

Source code

|  |
| --- |
| public List<AflEntry> AflEntries=new List<AflEntry>(); |

See Also

Applies to: [AflResult](#topic_0000000000000080)

CAKeyElement Class

Represents a CA key element

|  |  |
| --- | --- |
| C# |  |
| public class CAKeyElement | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Properties

[CAKey](#topic_0000000000000005), [Expiry](#topic_0000000000000008), [Exponent](#topic_0000000000000006), [Index](#topic_0000000000000004), [RID](#topic_0000000000000003), [SHA1Hash](#topic_0000000000000007)

CAKeyElement.CAKey Property

CaKey

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) CAKey {get; set;} | |

Source code

|  |
| --- |
| public string CAKey { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeyElement.Expiry Property

Certificate eχpiry date
Ημερομηνία λήξης πιστοποιητικού

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Expiry {get; set;} | |

Source code

|  |
| --- |
| public string Expiry { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeyElement.Exponent Property

Ca Key exponent
Εκθέτης κλειδιού Ca

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Exponent {get; set;} | |

Source code

|  |
| --- |
| public string Exponent { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeyElement.Index Property

CA key index
Δείκτης κλειδιού CA

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Index {get; set;} | |

Source code

|  |
| --- |
| public string Index { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeyElement.RID Property

Emv Application AID

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) RID {get; set;} | |

Source code

|  |
| --- |
| public string RID { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeyElement.SHA1Hash Property

SHA1 hash of the certificate
SHA1 hash του πιστοποιητικού

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) SHA1Hash {get; set;} | |

Source code

|  |
| --- |
| public string SHA1Hash { get; set; } |

See Also

Applies to: [CAKeyElement](#topic_0000000000000002)

CAKeys Class

Class that holds the known CA keys from file cakeys.xml

|  |  |
| --- | --- |
| C# |  |
| public class CAKeys | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Properties

[CaKeys](#topic_000000000000000A)

CAKeys.CaKeys Property

Ca keys array.

[System.Xml.Serialization is used to parse the xml file cakeys.xml

|  |  |
| --- | --- |
| C# |  |
| [System.Xml.Serialization.XmlElement("CAKeyElement")] public [CAKeyElement](#topic_0000000000000002)[] CaKeys {get; set;} | |

Source code

|  |
| --- |
| [System.Xml.Serialization.XmlElement("CAKeyElement")]  public CAKeyElement[] CaKeys { get; set; } |

See Also

Applies to: [CAKeys](#topic_0000000000000009)

CaKey Class

Το αντικείμενο περιέχει τις πληροφορίες που χρειάζονται για να γίνει Validate το certificate της κάρτας.

|  |  |
| --- | --- |
| C# |  |
| public class CaKey | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Properties

[Expiry](#topic_0000000000000013), [Exponent](#topic_0000000000000011), [Index](#topic_000000000000000F), [Key](#topic_0000000000000010), [Rid](#topic_000000000000000E), [SHA1Hash](#topic_0000000000000012)

CaKey.Expiry Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Expiry {get; set;} | |

Source code

|  |
| --- |
| public string Expiry { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKey.Exponent Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Exponent {get; set;} | |

Source code

|  |
| --- |
| public string Exponent { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKey.Index Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Index {get; set;} | |

Source code

|  |
| --- |
| public string Index { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKey.Key Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Key {get; set;} | |

Source code

|  |
| --- |
| public string Key { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKey.Rid Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Rid {get; set;} | |

Source code

|  |
| --- |
| public string Rid { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKey.SHA1Hash Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) SHA1Hash {get; set;} | |

Source code

|  |
| --- |
| public string SHA1Hash { get; set; } |

See Also

Applies to: [CaKey](#topic_000000000000000D)

CaKeyStore Class

Helper class that acting as a key storage manager.

|  |  |
| --- | --- |
| C# |  |
| public static class CaKeyStore | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Methods

[GetCaKey](#topic_000000000000000C)

CaKeyStore.GetCaKey Method

Retreives a CaKey from cakeys.xml

|  |  |
| --- | --- |
| C# |  |
| public static [CaKey](#topic_000000000000000D) GetCaKey(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *rid*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *index* ) | |

Parameters

rid

|  |
| --- |
|  |

index

|  |
| --- |
|  |

Returns

The CaKey object. Γίνεται δημιουργία ενός object CaKey, το οποίο επιστρέφεται.

Source code

|  |
| --- |
| public static CaKey GetCaKey(string rid, string index)  {    XElement root = XElement.Parse(Properties.Resources.cakeys);  var kkk=root.Elements("CAKeyElement");  var xel = kkk  .Where(x => x.Element("RID").Value == rid && x.Element("Index").Value == index).Select(c =>  new CaKey()  {  Rid = c.Element("RID").Value,  Expiry = c.Element("Expiry").Value,  Exponent = c.Element("Exponent").Value,  Key = c.Element("CAKey").Value,  Index = c.Element("Index").Value,  SHA1Hash = c.Element("SHA1Hash").Value,    }    ).SingleOrDefault();    return xel;  } |

See Also

Applies to: [CaKeyStore](#topic_000000000000000B)

CardNotSupportedExceprion Class

Exception thrown when the card is not supported by the current version of the application

|  |  |
| --- | --- |
| C# |  |
| public class CardNotSupportedExceprion : [ApplicationException](http://msdn.microsoft.com/en-us/library/system.applicationexception.aspx) | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[CardNotSupportedExceprion](#topic_0000000000000037)

CardNotSupportedExceprion Constructor

|  |  |
| --- | --- |
| C# |  |
| public CardNotSupportedExceprion() | |

Source code

|  |
| --- |
| public CardNotSupportedExceprion()  {  } |

See Also

Applies to: [CardNotSupportedExceprion](#topic_0000000000000036)

CardNotSupportedExceprion(String, Exception) Constructor

|  |  |
| --- | --- |
| C# |  |
| public CardNotSupportedExceprion(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *message*,  [Exception](http://msdn.microsoft.com/en-us/library/system.exception.aspx) *innerException* ) | |

Parameters

message

|  |
| --- |
|  |

innerException

|  |
| --- |
|  |

Source code

|  |
| --- |
| public CardNotSupportedExceprion(string message, Exception innerException) : base(message, innerException)  {  } |

See Also

Applies to: [CardNotSupportedExceprion](#topic_0000000000000036)

CardNotSupportedExceprion(String) Constructor

|  |  |
| --- | --- |
| C# |  |
| public CardNotSupportedExceprion(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *message* ) | |

Parameters

message

|  |
| --- |
|  |

Source code

|  |
| --- |
| public CardNotSupportedExceprion(string message) : base(message)  {  } |

See Also

Applies to: [CardNotSupportedExceprion](#topic_0000000000000036)

EmvConstants Class

Emv Constants used during a transaction.

|  |  |
| --- | --- |
| C# |  |
| public static class EmvConstants | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Methods

[getTagDescription](#topic_0000000000000035)

Structures

[AID](#topic_0000000000000023)

Enumerations

[EmvOfflineAuthType](#topic_0000000000000031), [GpoTemplateFormat](#topic_000000000000002C), [ResponceType](#topic_0000000000000027)

Fields

[AID\_LIST](#topic_0000000000000022), [AID\_LIST\_BYTES](#topic_0000000000000021), [APDU\_RES\_DESCRIPTION](#topic_000000000000002F), [ApplicationClass](#topic_0000000000000016), [ConstructedDataObject](#topic_000000000000001A), [ContextSpecificClass](#topic_0000000000000017), [PPSE](#topic_000000000000001D), [PSE](#topic_000000000000001C), [PdolTags](#topic_0000000000000020), [PrimitiveDataObject](#topic_0000000000000019), [PrivateClass](#topic_0000000000000018), [SELECT\_APDU\_HEADER](#topic_000000000000001E), [SW\_SELECT\_OK](#topic_000000000000001F), [SeeSubsequentBytes](#topic_000000000000001B), [UniversalClass](#topic_0000000000000015), [tags](#topic_0000000000000030)

EmvConstants.getTagDescription Method

Retreives the tag description for the given tag name

|  |  |
| --- | --- |
| C# |  |
| public static [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) getTagDescription(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *tagname* ) | |

Parameters

tagname

|  |
| --- |
| The tag name |

Returns

The tag Description or "Unknown EmvTag"

Source code

|  |
| --- |
| public static string getTagDescription(String tagname)  {    if (tags.ContainsKey(tagname.ToLower()))  {  return tags[tagname.ToLower()];    }  return "Unknown EmvTag";    } |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

AID Structure

EMV AID struct

|  |  |
| --- | --- |
| C# |  |
| public struct AID | |

Source code

|  |
| --- |
| public struct AID  {  public static readonly byte[] AID\_MASTERCARD\_CLASSIC = { 0xA0, 00, 00, 00, 04, 10, 10, 00 };  public static readonly byte[] AID\_VISA\_CLASSIC = { 0xA0, 00, 00, 00, 03, 10, 10, 00 };  public static readonly byte[] AID\_VISA\_ELECTRON = { 0xA0, 00, 00, 00, 03, 20, 10, 00 };      } |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Fields

[AID\_MASTERCARD\_CLASSIC](#topic_0000000000000024), [AID\_VISA\_CLASSIC](#topic_0000000000000025), [AID\_VISA\_ELECTRON](#topic_0000000000000026)

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

AID\_MASTERCARD\_CLASSIC Field

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] AID\_MASTERCARD\_CLASSIC | |

Source code

|  |
| --- |
| public static readonly byte[] AID\_MASTERCARD\_CLASSIC = { 0xA0, 00, 00, 00, 04, 10, 10, 00 }; |

See Also

Applies to: [AID](#topic_0000000000000023)

AID\_VISA\_CLASSIC Field

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] AID\_VISA\_CLASSIC | |

Source code

|  |
| --- |
| public static readonly byte[] AID\_VISA\_CLASSIC = { 0xA0, 00, 00, 00, 03, 10, 10, 00 }; |

See Also

Applies to: [AID](#topic_0000000000000023)

AID\_VISA\_ELECTRON Field

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] AID\_VISA\_ELECTRON | |

Source code

|  |
| --- |
| public static readonly byte[] AID\_VISA\_ELECTRON = { 0xA0, 00, 00, 00, 03, 20, 10, 00 }; |

See Also

Applies to: [AID](#topic_0000000000000023)

EmvOfflineAuthType Enumeration

Emv offline types

|  |  |  |
| --- | --- | --- |
| Constant | Value | Description |
| Cda | 256 | Combined data authentication |
| Dda | 512 | Dynamic data authentication |
| Sda | 1024 | Static data authentication |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

GpoTemplateFormat Enumeration

Emv Template format.

Emv book 3. Table 33: Data Elements Dictionary

|  |  |  |
| --- | --- | --- |
| Constant | Value | Description |
| Format1 | 128 | Response Message Template Format 1 |
| Format2 | 119 | Response Message Template Format 2 |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

ResponceType Enumeration

List of reponce types

|  |  |  |
| --- | --- | --- |
| Constant | Value | Description |
| All | 3 | Data from all reponse types |
| Gpo | 1 | Get processing options reponse type |
| ReaderRecord | 2 | Read record reponse type |
| Select | 0 | Select responce type |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

AID\_LIST Field

List of known Emv Applications in string format

|  |  |
| --- | --- |
| C# |  |
| new public static readonly [Dictionary](http://msdn.microsoft.com/en-us/library/xfhwa508.aspx)<[string](http://msdn.microsoft.com/en-us/library/system.string.aspx), [string](http://msdn.microsoft.com/en-us/library/system.string.aspx)> AID\_LIST | |

Source code

|  |
| --- |
| public static readonly Dictionary<string,string> AID\_LIST=new Dictionary<string, string>()  {  { "VISA CREDIT","A0000000031010"},  { "VISA DEBIT","A0000000032010"},  { "MASTERCARD CREDIT ","A0000000041010"},  { "MAESTRO CREDIT ","A0000000043060"},  { "AMEX","A0000000250000"},  }; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

AID\_LIST\_BYTES Field

List of known Emv Applications

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[][] AID\_LIST\_BYTES | |

Source code

|  |
| --- |
| public static readonly byte[][] AID\_LIST\_BYTES =  {  new byte[] {0xA0, 0x00, 0x00, 0x00,0x04,0x10,0x10},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x03,0x10,0x10},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x03,0x20,0x10},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x04,0x30,0x60},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x25,0x00,0x00},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x25,0x01},  new byte[] {0xA0, 0x00, 0x00, 0x00,0x25,0x01,0x01,0x04},  }; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

APDU\_RES\_DESCRIPTION Field

Emv known responces

|  |  |
| --- | --- |
| C# |  |
| new public static readonly [Dictionary](http://msdn.microsoft.com/en-us/library/xfhwa508.aspx)<[byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[], [string](http://msdn.microsoft.com/en-us/library/system.string.aspx)> APDU\_RES\_DESCRIPTION | |

Source code

|  |
| --- |
| public static readonly Dictionary<byte[],string> APDU\_RES\_DESCRIPTION=new Dictionary<byte[], string>()  {  {new byte[]{0x62,0x00},"No information given (NV-Ram not changed)"},  {new byte[]{0x62,0x01},"NV-Ram not changed 1."},  {new byte[]{0x62,0x81},"Part of returned data may be corrupted"},  {new byte[]{0x62,0x82},"End of file/record reached before reading Le bytes"},  {new byte[]{0x62,0x83},"Selected file invalidated"},  {new byte[]{0x62,0x84},"Selected file is not valid. FCI not formated according to ISO"},  {new byte[]{0x62,0x85},"No input data available from a sensor on the card. No Purse Engine enslaved for R3bc"},  {new byte[]{0x62,0xA2},"Wrong R-MAC"},  {new byte[]{0x62,0xA4},"Card locked (during reset( ))"},  {new byte[]{0x62,0xF1},"Wrong C-MAC"},  {new byte[]{0x62,0xF3},"Internal reset"},  {new byte[]{0x62,0xF5},"Default agent locked"},  {new byte[]{0x62,0xF7},"Cardholder locked"},  {new byte[]{0x62,0xF8},"Basement is current agent"},  {new byte[]{0x62,0xF9},"CALC Key Set not unblocked"},  {new byte[]{0x63,0x00},"No information given (NV-Ram changed)"},  {new byte[]{0x63,0x81},"File filled up by the last write. Loading/updating is not allowed."},  {new byte[]{0x63,0x82},"Card key not supported."},  {new byte[]{0x63,0x83},"Reader key not supported."},  {new byte[]{0x63,0x84},"Plaintext transmission not supported."},  {new byte[]{0x63,0x85},"Secured transmission not supported."},  {new byte[]{0x63,0x86},"Volatile memory is not available."},  {new byte[]{0x63,0x87},"Non-volatile memory is not available."},  {new byte[]{0x63,0x88},"Key number not valid."},  {new byte[]{0x63,0x89},"Key length is not correct."},  {new byte[]{0x63,0xC0},"Verify fail, no try left."},  {new byte[]{0x63,0xC1},"Verify fail, 1 try left."},  {new byte[]{0x63,0xC2},"Verify fail, 2 tries left."},  {new byte[]{0x63,0xC3},"Verify fail, 3 tries left."},  {new byte[]{0x64,0x00},"No information given (NV-Ram not changed)"},  {new byte[]{0x64,0x01},"Command timeout. Immediate response required by the card."},  {new byte[]{0x65,0x00},"No information given"},  {new byte[]{0x65,0x01},"Write error. Memory failure. There have been problems in writing or reading the EEPROM. Other hardware problems may also bring this error."},  {new byte[]{0x65,0x81},"Memory failure"},  {new byte[]{0x66,0x00},"Error while receiving (timeout)"},  {new byte[]{0x66,0x01},"Error while receiving (character parity error)"},  {new byte[]{0x66,0x02},"Wrong checksum"},  {new byte[]{0x66,0x03},"The current DF file without FCI"},  {new byte[]{0x66,0x04},"No SF or KF under the current DF"},  {new byte[]{0x66,0x69},"Incorrect Encryption/Decryption Padding"},  {new byte[]{0x67,0x00},"Wrong length"},  {new byte[]{0x68,0x00},"No information given (The request function is not supported by the card)"},  {new byte[]{0x68,0x81},"Logical channel not supported"},  {new byte[]{0x68,0x82},"Secure messaging not supported"},  {new byte[]{0x68,0x83},"Last command of the chain expected"},  {new byte[]{0x68,0x84},"Command chaining not supported"},  {new byte[]{0x69,0x00},"No information given (Command not allowed)"},  {new byte[]{0x69,0x01},"Command not accepted (inactive state)"},  {new byte[]{0x69,0x81},"Command incompatible with file structure"},  {new byte[]{0x69,0x82},"Security condition not satisfied."},  {new byte[]{0x69,0x83},"Authentication method blocked"},  {new byte[]{0x69,0x84},"Referenced data reversibly blocked (invalidated)"},  {new byte[]{0x69,0x85},"Conditions of use not satisfied."},  {new byte[]{0x69,0x86},"Command not allowed (no current EF)"},  {new byte[]{0x69,0x87},"Expected secure messaging (SM) object missing"},  {new byte[]{0x69,0x88},"Incorrect secure messaging (SM) data object"},  {new byte[]{0x69,0x8D},"Reserved"},  {new byte[]{0x69,0x96},"Data must be updated again"},  {new byte[]{0x69,0xE1},"POL1 of the currently Enabled Profile prevents this action."},  {new byte[]{0x69,0xF0},"Permission Denied"},  {new byte[]{0x69,0xF1},"Permission Denied - Missing Privilege"},  {new byte[]{0x6A,0x00},"No information given (Bytes P1 and/or P2 are incorrect)"},  {new byte[]{0x6A,0x80},"The parameters in the data field are incorrect."},  {new byte[]{0x6A,0x81},"Function not supported"},  {new byte[]{0x6A,0x82},"File not found"},  {new byte[]{0x6A,0x83},"Record not found"},  {new byte[]{0x6A,0x84},"There is insufficient memory space in record or file"},  {new byte[]{0x6A,0x85},"Lc inconsistent with TLV structure"},  {new byte[]{0x6A,0x86},"Incorrect P1 or P2 parameter."},  {new byte[]{0x6A,0x87},"Lc inconsistent with P1-P2"},  {new byte[]{0x6A,0x88},"Referenced data not found"},  {new byte[]{0x6A,0x89},"File already exists"},  {new byte[]{0x6A,0x8A},"DF name already exists."},  {new byte[]{0x6A,0xF0},"Wrong parameter value"},  {new byte[]{0x6B,0x00},"Wrong parameter(s) P1-P2"},  {new byte[]{0x6C,0x00},"Incorrect P3 length."},  {new byte[]{0x6D,0x00},"Instruction code not supported or invalid"},  {new byte[]{0x6E,0x00},"Class not supported"},  {new byte[]{0x6F,0x00},"Command aborted - more exact diagnosis not possible (e.g., operating system error)."},  {new byte[]{0x6F,0xFF},"Card dead (overuse, …)"},  {new byte[]{0x90,0x00},"Command successfully executed (OK)."},  {new byte[]{0x90,0x04},"PIN not succesfully verified, 3 or more PIN tries left"},  {new byte[]{0x90,0x08},"Key/file not found"},  {new byte[]{0x90,0x80},"Unblock Try Counter has reached zero"},  {new byte[]{0x91,0x00},"OK"},  {new byte[]{0x91,0x01},"States.activity, States.lock Status or States.lockable has wrong value"},  {new byte[]{0x91,0x02},"Transaction number reached its limit"},  {new byte[]{0x91,0x0C},"No changes"},  {new byte[]{0x91,0x0E},"Insufficient NV-Memory to complete command"},  {new byte[]{0x91,0x1C},"Command code not supported"},  {new byte[]{0x91,0x1E},"CRC or MAC does not match data"},  {new byte[]{0x91,0x40},"Invalid key number specified"},  {new byte[]{0x91,0x7E},"Length of command string invalid"},  {new byte[]{0x91,0x9D},"Not allow the requested command"},  {new byte[]{0x91,0x9E},"Value of the parameter invalid"},  {new byte[]{0x91,0xA0},"Requested AID not present on PICC"},  {new byte[]{0x91,0xA1},"Unrecoverable error within application"},  {new byte[]{0x91,0xAE},"Authentication status does not allow the requested command"},  {new byte[]{0x91,0xAF},"Additional data frame is expected to be sent"},  {new byte[]{0x91,0xBE},"Out of boundary"},  {new byte[]{0x91,0xC1},"Unrecoverable error within PICC"},  {new byte[]{0x91,0xCA},"Previous Command was not fully completed"},  {new byte[]{0x91,0xCD},"PICC was disabled by an unrecoverable error"},  {new byte[]{0x91,0xCE},"Number of Applications limited to 28"},  {new byte[]{0x91,0xDE},"File or application already exists"},  {new byte[]{0x91,0xEE},"Could not complete NV-write operation due to loss of power"},  {new byte[]{0x91,0xF0},"Specified file number does not exist"},  {new byte[]{0x91,0xF1},"Unrecoverable error within file"},  {new byte[]{0x92,0x10},"Insufficient memory. No more storage available."},  {new byte[]{0x92,0x40},"Writing to EEPROM not successful."},  {new byte[]{0x93,0x01},"Integrity error"},  {new byte[]{0x93,0x02},"Candidate S2 invalid"},  {new byte[]{0x93,0x03},"Application is permanently locked"},  {new byte[]{0x94,0x00},"No EF selected."},  {new byte[]{0x94,0x01},"Candidate currency code does not match purse currency"},  {new byte[]{0x94,0x02},"Candidate amount too high"},  {new byte[]{0x94,0x02},"Address range exceeded."},  {new byte[]{0x94,0x03},"Candidate amount too low"},  {new byte[]{0x94,0x04},"FID not found, record not found or comparison pattern not found."},  {new byte[]{0x94,0x05},"Problems in the data field"},  {new byte[]{0x94,0x06},"Required MAC unavailable"},  {new byte[]{0x94,0x07},"Bad currency : purse engine has no slot with R3bc currency"},  {new byte[]{0x94,0x08},"R3bc currency not supported in purse engine"},  {new byte[]{0x94,0x08},"Selected file type does not match command."},  {new byte[]{0x95,0x80},"Bad sequence"},  {new byte[]{0x96,0x81},"Slave not found"},  {new byte[]{0x97,0x00},"PIN blocked and Unblock Try Counter is 1 or 2"},  {new byte[]{0x97,0x02},"Main keys are blocked"},  {new byte[]{0x97,0x04},"PIN not succesfully verified, 3 or more PIN tries left"},  {new byte[]{0x97,0x84},"Base key"},  {new byte[]{0x97,0x85},"Limit exceeded - C-MAC key"},  {new byte[]{0x97,0x86},"SM error - Limit exceeded - R-MAC key"},  {new byte[]{0x97,0x87},"Limit exceeded - sequence counter"},  {new byte[]{0x97,0x88},"Limit exceeded - R-MAC length"},  {new byte[]{0x97,0x89},"Service not available"},  {new byte[]{0x98,0x02},"No PIN defined."},  {new byte[]{0x98,0x04},"Access conditions not satisfied, authentication failed."},  {new byte[]{0x98,0x35},"ASK RANDOM or GIVE RANDOM not executed."},  {new byte[]{0x98,0x40},"PIN verification not successful."},  {new byte[]{0x98,0x50},"INCREASE or DECREASE could not be executed because a limit has been reached."},  {new byte[]{0x99,0x00},"1 PIN try left"},  {new byte[]{0x99,0x04},"PIN not succesfully verified, 1 PIN try left"},  {new byte[]{0x99,0x85},"Wrong status - Cardholder lock"},  {new byte[]{0x99,0x86},"Missing privilege"},  {new byte[]{0x99,0x87},"PIN is not installed"},  {new byte[]{0x99,0x88},"Wrong status - R-MAC state"},  {new byte[]{0x9A,0x00},"2 PIN try left"},  {new byte[]{0x9A,0x04},"PIN not succesfully verified, 2 PIN try left"},  {new byte[]{0x9A,0x71},"Wrong parameter value - Double agent AID"},  {new byte[]{0x9A,0x72},"Wrong parameter value - Double agent Type"},  {new byte[]{0x9D,0x05},"Incorrect certificate type"},  {new byte[]{0x9D,0x07},"Incorrect session data size"},  {new byte[]{0x9D,0x08},"Incorrect DIR file record size"},  {new byte[]{0x9D,0x09},"Incorrect FCI record size"},  {new byte[]{0x9D,0x0A},"Incorrect code size"},  {new byte[]{0x9D,0x10},"Insufficient memory to load application"},  {new byte[]{0x9D,0x11},"Invalid AID"},  {new byte[]{0x9D,0x12},"Duplicate AID"},  {new byte[]{0x9D,0x13},"Application previously loaded"},  {new byte[]{0x9D,0x14},"Application history list full"},  {new byte[]{0x9D,0x15},"Application not open"},  {new byte[]{0x9D,0x17},"Invalid offset"},  {new byte[]{0x9D,0x18},"Application already loaded"},  {new byte[]{0x9D,0x19},"Invalid certificate"},  {new byte[]{0x9D,0x1A},"Invalid signature"},  {new byte[]{0x9D,0x1B},"Invalid KTU"},  {new byte[]{0x9D,0x1D},"MSM controls not set"},  {new byte[]{0x9D,0x1E},"Application signature does not exist"},  {new byte[]{0x9D,0x1F},"KTU does not exist"},  {new byte[]{0x9D,0x20},"Application not loaded"},  {new byte[]{0x9D,0x21},"Invalid Open command data length"},  {new byte[]{0x9D,0x30},"Check data parameter is incorrect (invalid start address)"},  {new byte[]{0x9D,0x31},"Check data parameter is incorrect (invalid length)"},  {new byte[]{0x9D,0x32},"Check data parameter is incorrect (illegal memory check area)"},  {new byte[]{0x9D,0x40},"Invalid MSM Controls ciphertext"},  {new byte[]{0x9D,0x41},"MSM controls already set"},  {new byte[]{0x9D,0x42},"Set MSM Controls data length less than 2 bytes"},  {new byte[]{0x9D,0x43},"Invalid MSM Controls data length"},  {new byte[]{0x9D,0x44},"Excess MSM Controls ciphertext"},  {new byte[]{0x9D,0x45},"Verification of MSM Controls data failed"},  {new byte[]{0x9D,0x50},"Invalid MCD Issuer production ID"},  {new byte[]{0x9D,0x51},"Invalid MCD Issuer ID"},  {new byte[]{0x9D,0x52},"Invalid set MSM controls data date"},  {new byte[]{0x9D,0x53},"Invalid MCD number"},  {new byte[]{0x9D,0x54},"Reserved field error"},  {new byte[]{0x9D,0x55},"Reserved field error"},  {new byte[]{0x9D,0x56},"Reserved field error"},  {new byte[]{0x9D,0x57},"Reserved field error"},  {new byte[]{0x9D,0x60},"MAC verification failed"},  {new byte[]{0x9D,0x61},"Maximum number of unblocks reached"},  {new byte[]{0x9D,0x62},"Card was not blocked"},  {new byte[]{0x9D,0x63},"Crypto functions not available"},  {new byte[]{0x9D,0x64},"No application loaded"},  {new byte[]{0x9E,0x00},"PIN not installed"},  {new byte[]{0x9E,0x04},"PIN not succesfully verified, PIN not installed"},  {new byte[]{0x9F,0x00},"PIN blocked and Unblock Try Counter is 3"},  {new byte[]{0x9F,0x04},"PIN not succesfully verified, PIN blocked and Unblock Try Counter is 3"},  }; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

PPSE Field

PPSE Application Name

|  |  |
| --- | --- |
| C# |  |
| public static readonly [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) PPSE | |

Source code

|  |
| --- |
| public static readonly string PPSE = "2PAY.SYS.DDF01"; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

PSE Field

PSE Application Name

|  |  |
| --- | --- |
| C# |  |
| public static readonly [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) PSE | |

Source code

|  |
| --- |
| public static readonly string PSE = "1PAY.SYS.DDF01"; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

PdolTags Field

Dictionary to hold the pdol tags and values

|  |  |
| --- | --- |
| C# |  |
| new public static [Dictionary](http://msdn.microsoft.com/en-us/library/xfhwa508.aspx)<[string](http://msdn.microsoft.com/en-us/library/system.string.aspx), [string](http://msdn.microsoft.com/en-us/library/system.string.aspx)> PdolTags | |

Source code

|  |
| --- |
| public static Dictionary<string, string> PdolTags = new Dictionary<string, string>(); |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

SELECT\_APDU\_HEADER Field

SELECT COMMAND ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] SELECT\_APDU\_HEADER | |

Source code

|  |
| --- |
| public static readonly byte[] SELECT\_APDU\_HEADER = {00, 0xA4, 04, 00}; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

SW\_SELECT\_OK Field

Responce ok ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public static readonly [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] SW\_SELECT\_OK | |

Source code

|  |
| --- |
| public static readonly byte[] SW\_SELECT\_OK = {0x90, 0x00}; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

tags Field

Emv known tags

|  |  |
| --- | --- |
| C# |  |
| new public static [Dictionary](http://msdn.microsoft.com/en-us/library/xfhwa508.aspx)<[string](http://msdn.microsoft.com/en-us/library/system.string.aspx), [string](http://msdn.microsoft.com/en-us/library/system.string.aspx)> tags | |

Source code

|  |
| --- |
| public static Dictionary<string, string> tags = new Dictionary<string, string>()  {  {"4f","Application Identifier (AID)"},  {"50","Application Label"},  {"57","Track 2 Equivalent Data"},  {"5a","Application Primary Account Number (PAN)"},  {"61","Application Template"},  {"6f","File Control Information (FCI) Template"},  {"70","Record Template"},  {"77","Response Message Template Format 2"},  {"80","Response Message Template Format 1"},  {"82","Application Interchange Profile"},  {"83","Command Template"},  {"84","DF Name"},  {"86","Issuer Script CommaRawOutput= Falsend"},  {"87","Application Priority Indicator"},  {"88","Short File Identifier"},  {"8c","Card Risk Management Data Object List 1 (CDOL1)"},  {"8d","Card Risk Management Data Object List 2 (CDOL2)"},  {"8e","Cardholder Verification Method (CVM) List"},  {"8f","Certification Authority Public Key Index"},  {"93","Signed Static Application Data"},  {"94","Application File Locator"},  {"97","Transaction Certificate Data Object List (TDOL)"},  {"9d","Directory Definition File"},  {"a5","Proprietary Information"},  {"5f20","Cardholder Name"},  {"5f24","Application Expiration Date YYMMDD"},  {"5f25","Application Effective Date YYMMDD"},  {"5f28","Issuer Country Code"},  {"5f2d","Language Preference"},  {"5f30","Service Code"},  {"5f34","Application Primary Account Number (PAN) Sequence Number"},  {"5f50","Issuer URL"},  {"90","Issuer Public Key"},  {"92","Issuer Public Key Remainder"},  {"9f05","Application Discretionary Data"},  {"9f07","Application Usage Control"},  {"9f08","Application Version Number"},  {"9f0d","Issuer Action Code - Default"},  {"9f0e","Issuer Action Code - Denial"},  {"9f0f","Issuer Action Code - Online"},  {"9f10","Issuer Application Data (IAD)"},  {"9f11","Issuer Code Table Index"},  {"9f12","Application Preferred Name"},  {"9f14","Lower Consecutive Offline Limit"},  {"9f17","PIN Try Counter"},  {"9f1f","Track 1 Discretionary Data"},  {"9f20","Track 2 Discretionary Data"},  {"9f23","Upper Consecutive Offline Limit"},  {"9f26","Application Cryptogram"},  {"9f32","Issuer Public Key Exponent"},  {"9f38","Processing Options Data Object List (PDOL)"},  {"9f42","Application Currency Code"},  {"9f44","Application Currency Exponent"},  {"9f46","ICC Public Key Certificate"},  {"9f47","ICC Public Key Exponent"},  {"9f48","ICC Public Key Remainder"},  {"9f49","Dynamic Data Authentication Data Object List (DDOL)"},  {"9f4a","Static Data Authentication EmvTag List"},  {"9f4f","Log Format"},  {"9f51","Application Currency Code"},  {"9f52","Application Default Action (ADA)"},  {"9f53","Consecutive Transaction Counter International Limit (CTCIL)"},  {"9f54","Cumulative Total Transaction Amount Limit (CTTAL)"},  {"9f56","Issuer Authentication Indicator"},  {"9f57","Issuer Country Code"},  {"9f58","Consecutive Transaction Counter Limit (CTCL)"},  {"9f59","Consecutive Transaction Counter Upper Limit (CTCUL)"},  {"9f5a","Application Program Identifier (Program ID)"},  {"9f5d","Application Capabilities Information (FCI)"},  {"9f60","P3 Generated 3DES KEYS"},  {"9f5c","Cumulative Total Transaction Amount Upper Limit (CTTAUL)"},  {"9f5e","Consecutive Transaction International Upper Limit (CTIUL)"},  {"9f68","Card Additional Processes"},  {"9f69","Card Authentication Related Data"},  {"9f6c","Card Transaction Qualifiers (CTQ)"},  {"9f6e","Form Factor Indicator"},  {"9f72","Consecutive Transaction Limit (International-Country)"},  {"9f73","Currency Conversion Parameters"},  {"bf0c","File Control Information (FCI) Issuer Discretionary Data"},  {"bf55","Paywave Template VLP Funds"},  {"bf56","Paywave Template Consecutive Transaction"},  {"bf57","Paywave Template Consecutive Transaction International"},  {"bf58","Paywave Template Cumulative Total Transaction"},  {"bf5b","DF01 in BF5B -> Application Capabilities"},  {"c3","Card Issuer Action Code (Contact) - Decline"},  {"c4","Card Issuer Action Code (Contact) - Default"},  {"c5","Card Issuer Action Code (Contact) - Online"},  {"c6","PIN Try Limit"},  {"c7","CDOL1 Related Data Length"},  {"c8","CRM Country Code"},  {"c9","Accumulator 1 Currency Code"},  {"ca","Accumulator 1 Lower Limit"},  {"cb","Accumulator 1 Upper Limit"},  {"d1","Accumulator 1 Currency Conversion Table"},  {"d3","Pin Try Limit"},  {"d5","Application Control (Contact)"},  {"d6","Default ARPC Response Code"},  {"de","Log Data Table"},  {"df01","Encrypted PIN Block in EmvTag 9F62 - ISO 95641 Format 0"},  {"df11","Accumulator 1 Control (Contact)"},  {"df13","DDA Public Modulus"},  {"df14","Accumulator 2 Control (Contact)"},  {"df15","DDA Public Modulus KCV"},  {"df16","Accumulator 2 Currency Code"},  {"df17","DDA Public Modulus Length EmvTag or Accumulator 2 Currency Conversion Table"},  {"df18","Accumulator 2 Lower Limit"},  {"df19","Accumulator 2 Upper Limit"},  {"df1a","Counter 1 Control (Contact)"},  {"df1d","Counter 2 Control (Contact)"},  {"df1f","Counter 2 Lower Limit"},  {"df21","Counter 2 Upper Limit"},  {"df22","MTA CVM (Contact)"},  {"df24","MTA Currency Code"},  {"df25","MTA NoCVM (Contact)"},  {"df27","Number Of Days Offline Limit"},  {"df28","Accumulator 1 CVR Dependency Data (Contact)"},  {"df2a","Accumulator 2 CVR Dependency Data (Contact)"},  {"df2c","Counter 1 CVR Dependency Data (Contact)"},  {"df2e","Counter 2 CVR Dependency Data (Contact)"},  {"df32","SMI Session Key Counter Limit (Contact)"},  {"df36","PIN Decipherments Error Counter Limit"},  {"df3a","AC Session Key Counter Limit (Contact)"},  {"df3c","CVR Issuer Discretionary Data (Contact)"},  {"df3f","Read Record Filter (Contact)"},  {"df60","DDA Component P"},  {"df61","DDA Component Q"},  {"df62","DDA Component D1"},  {"df63","DDA Component D2"},  {"df64","DDA Component Q Minus 1 Mod P"},  {"df70","DDA Component P"},  {"df71","DDA Component Q"},  {"df72","DDA Component D1"},  {"df73","DDA Component D2"},  {"df74","DDA Component Q Minus 1 Mod P"},    }; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

ApplicationClass Field

Universal class ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) ApplicationClass = 64 | |

Source code

|  |
| --- |
| public const byte ApplicationClass = 0x40; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

ConstructedDataObject Field

Constructed Data Object ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) ConstructedDataObject = 32 | |

Source code

|  |
| --- |
| public const byte ConstructedDataObject = 0x20; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

ContextSpecificClass Field

Universal class ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) ContextSpecificClass = 128 | |

Source code

|  |
| --- |
| public const byte ContextSpecificClass = 0x80; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

PrimitiveDataObject Field

Primitive Data Object ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) PrimitiveDataObject = 31 | |

Source code

|  |
| --- |
| public const byte PrimitiveDataObject = 0x1f; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

PrivateClass Field

Universal class ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) PrivateClass = 192 | |

Source code

|  |
| --- |
| public const byte PrivateClass = 0xc0; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

SeeSubsequentBytes Field

See Sub sequentBytes ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) SeeSubsequentBytes = 31 | |

Source code

|  |
| --- |
| public const byte SeeSubsequentBytes = 0x1f; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

UniversalClass Field

Universal class ISO 7816 part 4

|  |  |
| --- | --- |
| C# |  |
| public const [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx) UniversalClass = 31 | |

Source code

|  |
| --- |
| public const byte UniversalClass = 0x1f; |

See Also

Applies to: [EmvConstants](#topic_0000000000000014)

OfflineAuth Class

Class representing the Offline authorization of an emv smart card

|  |  |
| --- | --- |
| C# |  |
| public class OfflineAuth | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[OfflineAuth](#topic_000000000000003F)

Properties

[AuthType](#topic_0000000000000046)

Methods

[DecryptRsa](#topic_0000000000000043), [GetSha1](#topic_0000000000000045), [doCdaAuth](#topic_0000000000000040), [doDdaAuth](#topic_0000000000000041), [doSdaAuth](#topic_0000000000000042)

Fields

[ICC\_KEY\_HASH](#topic_000000000000003E)

OfflineAuth Constructor

Constractor storing the SmartApplication for subsequent use

|  |  |
| --- | --- |
| C# |  |
| public OfflineAuth(  [SmartApplication](#topic_0000000000000049) *app* ) | |

Parameters

app

|  |
| --- |
| SmartApplication initialized by the caller |

Source code

|  |
| --- |
| public OfflineAuth(SmartApplication app)  {  \_app = app;  switch (AuthType)  {  case EmvConstants.EmvOfflineAuthType.Cda:  doCdaAuth();  break;  default:  throw new NotImplementedException($"AuthType {AuthType} Not Implemented");  }      } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.AuthType Property

Retreives the type of authentication to use

|  |  |
| --- | --- |
| C# |  |
| public [EmvOfflineAuthType](#topic_0000000000000031) AuthType {get;} | |

Source code

|  |
| --- |
| public EmvConstants.EmvOfflineAuthType AuthType  {  get  {  var tagAip = \_app.GetTagValue(EmvConstants.ResponceType.All, "82");  if (string.IsNullOrEmpty(tagAip))  {  throw new CardNotSupportedExceprion("Card not supported");  }  var AipValue = int.Parse(tagAip, System.Globalization.NumberStyles.HexNumber);  bool exists = Enum.IsDefined(typeof(EmvConstants.EmvOfflineAuthType), AipValue);  if ((AipValue & (int)EmvConstants.EmvOfflineAuthType.Cda) != 0)  {  return EmvConstants.EmvOfflineAuthType.Cda;  }  if ((AipValue & (int)EmvConstants.EmvOfflineAuthType.Dda) != 0)  {  return EmvConstants.EmvOfflineAuthType.Dda;  }  if ((AipValue & (int)EmvConstants.EmvOfflineAuthType.Sda) != 0)  {  return EmvConstants.EmvOfflineAuthType.Sda;  }    throw new ApplicationException($"Failed to Parse Offline Auth type for AIP value 0x{tagAip} ");  }  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.DecryptRsa (String, String, String) Method

Decrypt data using RSA Asymmetric Block Cipher

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) DecryptRsa(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *data*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *exponent*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *key* ) | |

Parameters

data

|  |
| --- |
| The data to decrypt |

exponent

|  |
| --- |
| The key exponent to use |

key

|  |
| --- |
| The RSA key to use |

Returns

The decrypted data

Source code

|  |
| --- |
| public string DecryptRsa(string data, string exponent, string key)  {    BigInteger mod = new BigInteger(key, 16);  BigInteger pubExp = new BigInteger(exponent, 16);    RsaKeyParameters pubParameters = new RsaKeyParameters(false, mod, pubExp);  IAsymmetricBlockCipher eng = new RsaEngine();  eng.Init(true, pubParameters);  byte[] encdata = StringTools.HexStringToByteArray(data);  encdata = eng.ProcessBlock(encdata, 0, encdata.Length);  string result = StringTools.ByteArrayToHexString(encdata);  return result;    } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.DecryptRsa (String, String) Method

DecryptRsa overloading of DecryptRsa(string data, string exponent, string key)

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) DecryptRsa(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *data*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *exponent* ) | |

Parameters

data

|  |
| --- |
| The data to decrypt |

exponent

|  |
| --- |
| The key exponent to use |

Returns

The decrypted data

Source code

|  |
| --- |
| public string DecryptRsa(string data, string exponent)  {  return DecryptRsa(data, exponent, \_caKey.Key);  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.GetSha1 Method

Compute SHA1 hash for the given input data

|  |  |
| --- | --- |
| C# |  |
| public static [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) GetSha1(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *input* ) | |

Parameters

input

|  |
| --- |
|  |

Returns

Hash value

Source code

|  |
| --- |
| public static string GetSha1(string input)  {  using (var sha1 = System.Security.Cryptography.SHA1.Create())  {  byte[] inputBytes = StringTools.HexStringToByteArray(input);  byte[] hash = sha1.ComputeHash(inputBytes);    StringBuilder sb = new StringBuilder();  for (int i = 0; i < hash.Length; i++)  {  sb.Append(hash[i].ToString("X2"));  }  return sb.ToString();  }  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.doCdaAuth Method

Half implemeted. Falls back to DDA

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) doCdaAuth() | |

Source code

|  |
| --- |
| public void doCdaAuth()  {  BasicAuth();  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.doDdaAuth Method

Validates card using Dynamic data Authentication

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) doDdaAuth() | |

Source code

|  |
| --- |
| public void doDdaAuth()  {  BasicAuth();  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

OfflineAuth.doSdaAuth Method

**NOTE: This member is now obsolete.**

Please use DDA or CDA authentication

|  |  |
| --- | --- |
| C# |  |
| [Obsolete("Please use DDA or CDA authentication")] public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) doSdaAuth() | |

Source code

|  |
| --- |
| [Obsolete("Please use DDA or CDA authentication")]  public void doSdaAuth()  {  throw new NotImplementedException("Sda not");  } |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

ICC\_KEY\_HASH Field

|  |  |
| --- | --- |
| C# |  |
| public [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] ICC\_KEY\_HASH | |

Source code

|  |
| --- |
| public byte[] ICC\_KEY\_HASH; |

See Also

Applies to: [OfflineAuth](#topic_000000000000003D)

SCardReaderExtensions Class

Extends Trasmit command writing a log output to System.Diagnostics.Debug.WriteLine when Compiler Header DEBUG is true

|  |  |
| --- | --- |
| C# |  |
| public static class SCardReaderExtensions | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Methods

[TransmitWithLog](#topic_0000000000000048)

SCardReaderExtensions.TransmitWithLog Method

The implementation of transmit with log

|  |  |
| --- | --- |
| C# |  |
| public static **Response** TransmitWithLog(  this **IsoReader** *reader*,  **CommandApdu** *command* ) | |

Parameters

reader

|  |
| --- |
| Reader Name |

command

|  |
| --- |
| APDU command |

Source code

|  |
| --- |
| public static Response TransmitWithLog(this IsoReader reader, CommandApdu command)  {  SCardPCI receivePci = new SCardPCI(); // IO returned protocol control information.  IntPtr sendPci = SCardPCI.GetPci(reader.ActiveProtocol);        #if DEBUG    System.Diagnostics.Debug.WriteLine(StringTools.ByteArrayToHexString(command.ToArray()));  #endif    var res = reader.Transmit(command); // data buffer    #if DEBUG  System.Diagnostics.Debug.WriteLine(StringTools.ByteArrayToHexString(res.GetData()));  #endif    if (res.SW1 != 0x61) return res;  CommandApdu apdu2 = new CommandApdu(IsoCase.Case2Short, reader.ActiveProtocol)  {  CLA = new ClassByte(ClaHighPart.Iso0x, SecureMessagingFormat.None, 0),  Instruction = InstructionCode.GetResponse,  P1 = 0x00,  P2 = 00,  Le = res.SW2  };    #if DEBUG  System.Diagnostics.Debug.WriteLine(StringTools.ByteArrayToHexString(apdu2.ToArray()));  #endif  res = reader.Transmit(apdu2);  #if DEBUG   System.Diagnostics.Debug.WriteLine(StringTools.ByteArrayToHexString(res.GetData()));  #endif  return res;  } |

See Also

Applies to: [SCardReaderExtensions](#topic_0000000000000047)

SmartApplication Class

Object representation of Smart card Application

|  |  |
| --- | --- |
| C# |  |
| public class SmartApplication | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[SmartApplication](#topic_000000000000004A)

Properties

[AID](#topic_000000000000004C), [EmvRecords](#topic_0000000000000051), [GpoTemplateFormat](#topic_0000000000000052), [RES\_GPO](#topic_000000000000004E), [RES\_SELECT](#topic_000000000000004D), [TLV\_GPO](#topic_0000000000000050), [TLV\_SELECT](#topic_000000000000004F)

Methods

[Dump](#topic_0000000000000058), [GetOfflineTagValues](#topic_0000000000000057), [GetProcessingOptions](#topic_0000000000000054), [GetTagValue](#topic_0000000000000056), [ReadRecords](#topic_0000000000000055), [SelectApplication](#topic_0000000000000053), [SetGpo](#topic_000000000000004B)

SmartApplication Constructor

Constractor for the SmartApplication object

|  |  |
| --- | --- |
| C# |  |
| public SmartApplication(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *selectresponce*,  **IsoReader** *reader* ) | |

Parameters

selectresponce

|  |
| --- |
| The reponce of the card following the SELECT application command |

reader

|  |
| --- |
| The IsoReader communicating with the application/card |

Source code

|  |
| --- |
| public SmartApplication(byte[] selectresponce, IsoReader reader)  {  \_reader = reader;  RES\_SELECT = selectresponce;  TLV\_SELECT = new SmartTlv(selectresponce);  AID = TLV\_SELECT.TagList.Single(t => t.TagStringName == "84").TagValue.ToArray();  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.AID Property

The application Identifier in bytes

|  |  |
| --- | --- |
| C# |  |
| public [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] AID {get;} | |

Source code

|  |
| --- |
| public byte[] AID { get; private set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.EmvRecords Property

List of Emv records in the card

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[SmartEmvRecord](#topic_0000000000000062)> EmvRecords {get;} | |

Source code

|  |
| --- |
| public List<SmartEmvRecord> EmvRecords { get; private set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.GpoTemplateFormat Property

The Response Message Template Format of the GPO responce.

|  |  |
| --- | --- |
| C# |  |
| public [GpoTemplateFormat](#topic_000000000000002C) GpoTemplateFormat {get;} | |

Source code

|  |
| --- |
| public EmvConstants.GpoTemplateFormat GpoTemplateFormat { get; private set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.RES\_GPO Property

The reponce of the GPO command

|  |  |
| --- | --- |
| C# |  |
| public [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] RES\_GPO {get; set;} | |

Source code

|  |
| --- |
| public byte[] RES\_GPO { get; set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.RES\_SELECT Property

The reponce of the select command

|  |  |
| --- | --- |
| C# |  |
| public [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] RES\_SELECT {get; set;} | |

Source code

|  |
| --- |
| public byte[] RES\_SELECT { get; set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.TLV\_GPO Property

The reponce of the GPO command in Tag-Length-Value format

|  |  |
| --- | --- |
| C# |  |
| public [SmartTlv](#topic_0000000000000069) TLV\_GPO {get;} | |

Source code

|  |
| --- |
| public SmartTlv TLV\_GPO { get; private set; } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.TLV\_SELECT Property

The reponce of the select command in Tag-Length-Value format

|  |  |
| --- | --- |
| C# |  |
| public [SmartTlv](#topic_0000000000000069) TLV\_SELECT {get;} | |

Source code

|  |
| --- |
| public SmartTlv TLV\_SELECT { get;} |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.Dump Method

String representation of the class

|  |  |
| --- | --- |
| C# |  |
| public [StringBuilder](http://msdn.microsoft.com/en-us/library/system.text.stringbuilder.aspx) Dump() | |

Returns

The representation within a StringBuilder class

Source code

|  |
| --- |
| public StringBuilder Dump()  {  StringBuilder sb = new StringBuilder();  this.TLV\_GPO.dump(ref sb);  this.TLV\_SELECT.dump(ref sb);  foreach (var rc in this.EmvRecords)  {  rc.TLV\_READRECORD.dump(ref sb);  }    return sb;  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.GetOfflineTagValues Method

Retreive the values used for the Offline Auth

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) GetOfflineTagValues() | |

Returns

The values concatenated. Sda tag list included, if any

Source code

|  |
| --- |
| public string GetOfflineTagValues()  {  var offlineRecords = this.EmvRecords.Where(r => r.IsOffline);  string offlineTagsConcat = "";  foreach (var record in offlineRecords)  {  foreach (var tag in record.TLV\_READRECORD.TagList)  {  offlineTagsConcat += tag.TlvData;  }  }  var sdaTagList = GetTagValue(EmvConstants.ResponceType.ReaderRecord, "9F4A");  if (!string.IsNullOrEmpty(sdaTagList))  {  offlineTagsConcat += GetTagValue(EmvConstants.ResponceType.ReaderRecord, sdaTagList);  }      return offlineTagsConcat;  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.GetProcessingOptions Method

Issues a GPO command.

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) GetProcessingOptions() | |

Remarks

Pdol data , reponce parsing and storing are handled internaly

Source code

|  |
| --- |
| public void GetProcessingOptions()  {  List<byte> pdoldata=new List<byte> {0x83};  var pdol=TLV\_SELECT.TagList.SingleOrDefault(t => t.TagStringName == "9F38");  if (pdol != null)  {  var tempdata = TlvTools.parseTagLengthData(pdol.TagValue.ToArray());  pdoldata.Add((byte) tempdata.Length);  pdoldata.AddRange(tempdata);  }  else  {  pdoldata.Add(0x00);  }    CommandApdu apdu = new CommandApdu(IsoCase.Case4Short, \_reader.ActiveProtocol)  {  CLA = new ClassByte(ClaHighPart.Iso8x, SecureMessagingFormat.None, 0),  INS = 0xA8,  P1 = 0x00,  P2 = 00,  Data = pdoldata.ToArray()  };    var res = \_reader.TransmitWithLog(apdu); // data buffer      RES\_GPO = res.GetData();  GpoTemplateFormat = (RES\_GPO[0] == (byte) EmvConstants.GpoTemplateFormat.Format1)  ? EmvConstants.GpoTemplateFormat.Format1  : EmvConstants.GpoTemplateFormat.Format2;  TLV\_GPO = new SmartTlv(RES\_GPO);  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.GetTagValue Method

Retreive the value of the gigen tag

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) GetTagValue(  [ResponceType](#topic_0000000000000027) *type*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *tagname* ) | |

Parameters

type

|  |
| --- |
| Where to look in the card data |

tagname

|  |
| --- |
| The tag name |

Returns

The tag data

Source code

|  |
| --- |
| public string GetTagValue(EmvConstants.ResponceType type, string tagname)  {  string res = string.Empty;  switch (type)  {  case EmvConstants.ResponceType.All:  res = \_GetTagValue(EmvConstants.ResponceType.ReaderRecord, tagname);  if (string.IsNullOrEmpty(res))  {  res = \_GetTagValue(EmvConstants.ResponceType.Gpo, tagname);  }  return res;  case EmvConstants.ResponceType.Select:  throw new NotImplementedException("EMV.ResponceType.Select not implemented");  default:  return \_GetTagValue(type, tagname);    }  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.ReadRecords Method

Issue a read-record command for all records included in the GPO Responce, Tag AFL(Application file locator)

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) ReadRecords() | |

Source code

|  |
| --- |
| public void ReadRecords()  {  byte[] afl;  if (GpoTemplateFormat == EmvConstants.GpoTemplateFormat.Format1)  {  List<byte>gpores = TLV\_GPO.TagList.Single(t => t.TagStringName == "80").TagValue;  afl = gpores.Skip(2).ToArray();  }  else  {  afl = TLV\_GPO.TagList.SingleOrDefault(t => t.TagStringName == "94")?.TagValue.ToArray();  if (afl == null)  {  return;  }  }    AflResult aflpos= TlvTools.AflParser(afl);  EmvRecords = new List<SmartEmvRecord>();  foreach (var entry in aflpos.AflEntries)  {  for (int irecord= entry.StartRecord;irecord<=entry.EndRecord;irecord++ )  //foreach (int irecord in Enumerable.Range(entry.StartRecord, entry.EndRecord))  {  CommandApdu apdu = new CommandApdu(IsoCase.Case2Short, \_reader.ActiveProtocol);  apdu.CLA = 0x00;  apdu.INS = 0xB2; //GPO  apdu.P1 =(byte) irecord; //select by name  apdu.P2 = (byte) ((entry.Sfi << 3) | 4); // First or only occurrence    var res = \_reader.TransmitWithLog(apdu); // data buffer    if (res.SW1 == 0x6C)  {  apdu.Le = res.SW2;  res = \_reader.TransmitWithLog(apdu); // data buffer  }    if (res.SW1 != (byte)SW1Code.Normal)  {  throw new PCSCException(SCardError.CardUnsupported, "GPO not fully supported");  }  var record = new SmartEmvRecord(entry.Sfi,irecord,(irecord<=entry.OfflineRecords), res.GetData());  EmvRecords.Add(record);    }  }  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.SelectApplication Method

Issues a select Application command

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) SelectApplication() | |

Source code

|  |
| --- |
| public void SelectApplication()  {  //\_reader.Reconnect(SCardShareMode.Exclusive, SCardProtocol.Any, SCardReaderDisposition.Eject);  CommandApdu apdu = new CommandApdu(IsoCase.Case4Short, \_reader.ActiveProtocol)  {  CLA = new ClassByte(ClaHighPart.Iso0x, SecureMessagingFormat.None, 0),  Instruction = InstructionCode.SelectFile,  P1 = 0x04,  P2 = 00,  Data = AID  };    var res = \_reader.TransmitWithLog(apdu);  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartApplication.SetGpo Method

Prepares the Get processing options command

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) SetGpo(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *gpodata* ) | |

Parameters

gpodata

|  |
| --- |
|  |

Source code

|  |
| --- |
| public void SetGpo(byte[] gpodata)  {  RES\_GPO = gpodata;  TLV\_GPO = new SmartTlv(gpodata);  } |

See Also

Applies to: [SmartApplication](#topic_0000000000000049)

SmartCard Class

|  |  |
| --- | --- |
| C# |  |
| public class SmartCard : [IDisposable](http://msdn.microsoft.com/en-us/library/system.idisposable.aspx) | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[SmartCard](#topic_000000000000005C)

Properties

[Applications](#topic_000000000000005A)

Methods

[Dispose](#topic_0000000000000061), [GetAIDs](#topic_0000000000000060), [GetSingleApplication](#topic_000000000000005F)

Fields

[reader](#topic_000000000000005B)

SmartCard(String) Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartCard(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *readerName* ) | |

Parameters

readerName

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartCard(string readerName)  {  var contextFactory = ContextFactory.Instance;  \_context = contextFactory.Establish(SCardScope.System);  if (string.IsNullOrEmpty(readerName))  {  throw new ApplicationException("No smartCard readers found");  }    reader = new IsoReader(\_context, readerName, SCardShareMode.Exclusive, SCardProtocol.Any, true);    } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard(SCardContext, String) Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartCard(  **SCardContext** *context*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *readerName* ) | |

Parameters

context

|  |
| --- |
|  |

readerName

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartCard(SCardContext context, string readerName)  {  reader = new IsoReader(context);  reader.Connect(readerName, SCardShareMode.Exclusive, SCardProtocol.Any);      } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard(IsoReader) Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartCard(  **IsoReader** *reader* ) | |

Parameters

reader

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartCard(IsoReader reader)  {  this.reader = reader;  } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard.Applications Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[SmartApplication](#topic_0000000000000049)> Applications {get; set;} | |

Source code

|  |
| --- |
| public List<SmartApplication> Applications { get; set; } = new List<SmartApplication>(); |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard.Dispose Method

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) Dispose() | |

Source code

|  |
| --- |
| public void Dispose()  {  reader.Disconnect(SCardReaderDisposition.Unpower);  //reader.CurrentContext.Release();  //reader.CurrentContext.Dispose();        } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard.GetAIDs Method

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) GetAIDs(  [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) *ReturnOnFirst* = false ) | |

Parameters

ReturnOnFirst

|  |
| --- |
|  |

Source code

|  |
| --- |
| public int GetAIDs(bool ReturnOnFirst=false)  {    CommandApdu apdu = new CommandApdu(IsoCase.Case4Short, reader.ActiveProtocol);  apdu.CLA = new ClassByte(ClaHighPart.Iso0x, SecureMessagingFormat.None, 0);  apdu.Instruction = InstructionCode.SelectFile;  apdu.P1 = 0x04; //select by name  apdu.P2 = 00; // First or only occurrence  //apdu.Le = 0;  foreach (var app in EmvConstants.AID\_LIST\_BYTES)  {  byte[] aid = (app);  apdu.Data = aid;  var res = reader.Transmit(apdu); // data buffer  if (res.SW1 == 0x90)  {  Applications.Add(new SmartApplication(res.GetData(),reader));  if (ReturnOnFirst)  {  return 1;  }  }  }  return Applications.Count;  } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartCard.GetSingleApplication Method

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) GetSingleApplication(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *aid* ) | |

Parameters

aid

|  |
| --- |
|  |

Source code

|  |
| --- |
| public void GetSingleApplication(byte[] aid)  {    CommandApdu apdu = new CommandApdu(IsoCase.Case4Short, reader.ActiveProtocol);  apdu.CLA = new ClassByte(ClaHighPart.Iso0x, SecureMessagingFormat.None, 0);  apdu.Instruction = InstructionCode.SelectFile;  apdu.P1 = 0x04; //select by name  apdu.P2 = 00; // First or only occurrence  apdu.Data = aid;  System.Diagnostics.Debug.WriteLine(StringTools.ByteArrayToHexString(apdu.ToArray()));  Response res = reader.Transmit(  apdu);      if (res.SW1 == 0x90)  {  Applications.Add(new SmartApplication(res.GetData(), reader));  }  else  {  throw new PCSCException(SCardError.FileNotFound, "Select command failed");  }  } |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

reader Field

|  |  |
| --- | --- |
| C# |  |
| public **IsoReader** reader | |

Source code

|  |
| --- |
| public IsoReader reader; |

See Also

Applies to: [SmartCard](#topic_0000000000000059)

SmartEmvRecord Class

|  |  |
| --- | --- |
| C# |  |
| public class SmartEmvRecord | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[SmartEmvRecord](#topic_0000000000000068)

Properties

[IsOffline](#topic_0000000000000065), [RES\_READRECORD](#topic_0000000000000066), [Record](#topic_0000000000000064), [Sfi](#topic_0000000000000063), [TLV\_READRECORD](#topic_0000000000000067)

SmartEmvRecord Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartEmvRecord(  [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) *sfi*,  [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) *record*,  [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) *isoffline*,  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *data* ) | |

Parameters

sfi

|  |
| --- |
|  |

record

|  |
| --- |
|  |

isoffline

|  |
| --- |
|  |

data

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartEmvRecord(int sfi, int record, bool isoffline, byte[] data)  {  Sfi = sfi;  Record = record;  IsOffline = isoffline;  RES\_READRECORD = data;  TLV\_READRECORD = new SmartTlv(data);  } |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartEmvRecord.IsOffline Property

|  |  |
| --- | --- |
| C# |  |
| public [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) IsOffline {get;} | |

Source code

|  |
| --- |
| public bool IsOffline { get; } |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartEmvRecord.RES\_READRECORD Property

|  |  |
| --- | --- |
| C# |  |
| public [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] RES\_READRECORD {get;} | |

Source code

|  |
| --- |
| public byte[] RES\_READRECORD { get; } |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartEmvRecord.Record Property

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) Record {get;} | |

Source code

|  |
| --- |
| public int Record { get; } |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartEmvRecord.Sfi Property

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) Sfi {get;} | |

Source code

|  |
| --- |
| public int Sfi { get; } |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartEmvRecord.TLV\_READRECORD Property

|  |  |
| --- | --- |
| C# |  |
| public [SmartTlv](#topic_0000000000000069) TLV\_READRECORD {get;} | |

Source code

|  |
| --- |
| public SmartTlv TLV\_READRECORD { get;} |

See Also

Applies to: [SmartEmvRecord](#topic_0000000000000062)

SmartTag Class

|  |  |
| --- | --- |
| C# |  |
| public class SmartTag | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[SmartTag](#topic_000000000000007A)

Properties

[Children](#topic_0000000000000078), [DataLen](#topic_0000000000000072), [Description](#topic_0000000000000074), [HasChildren](#topic_0000000000000076), [HasParent](#topic_0000000000000075), [IsConstructed](#topic_0000000000000079), [Parent](#topic_0000000000000077), [TagStringName](#topic_0000000000000070), [TagValue](#topic_0000000000000073), [Tagname](#topic_000000000000006F), [TlvData](#topic_0000000000000071)

SmartTag Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartTag(  [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)> *tagname*,  [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) *dataLen*,  [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)> *tagValue*,  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *tagDescript*,  [SmartTag](#topic_000000000000006E) *parent* = null ) | |

Parameters

tagname

|  |
| --- |
|  |

dataLen

|  |
| --- |
|  |

tagValue

|  |
| --- |
|  |

tagDescript

|  |
| --- |
|  |

parent

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartTag(List<byte> tagname, int dataLen, List<byte> tagValue, string tagDescript,SmartTag parent=null)  {  Tagname = tagname;  DataLen = dataLen;  TagValue = tagValue;  Description = tagDescript;  Parent = parent;  } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.Children Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[SmartTag](#topic_000000000000006E)> Children {get; set;} | |

Source code

|  |
| --- |
| public List<SmartTag> Children { get; set; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.DataLen Property

|  |  |
| --- | --- |
| C# |  |
| public [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) DataLen {get;} | |

Source code

|  |
| --- |
| public int DataLen { get; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.Description Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) Description {get;} | |

Source code

|  |
| --- |
| public string Description { get; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.HasChildren Property

|  |  |
| --- | --- |
| C# |  |
| public [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) HasChildren {get;} | |

Source code

|  |
| --- |
| public bool HasChildren => Children != null && Children.Any(); |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.HasParent Property

|  |  |
| --- | --- |
| C# |  |
| public [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) HasParent {get;} | |

Source code

|  |
| --- |
| public bool HasParent => Parent != null; |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.IsConstructed Property

|  |  |
| --- | --- |
| C# |  |
| public [bool](http://msdn.microsoft.com/en-us/library/system.boolean.aspx) IsConstructed {get;} | |

Source code

|  |
| --- |
| public bool IsConstructed => (Tagname[0] & EmvConstants.ConstructedDataObject) == EmvConstants.ConstructedDataObject; |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.Parent Property

|  |  |
| --- | --- |
| C# |  |
| public [SmartTag](#topic_000000000000006E) Parent {get; set;} | |

Source code

|  |
| --- |
| public SmartTag Parent { get; set; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.TagStringName Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) TagStringName {get;} | |

Source code

|  |
| --- |
| public string TagStringName => StringTools.ByteArrayToHexString(Tagname.ToArray()); |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.TagValue Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)> TagValue {get;} | |

Source code

|  |
| --- |
| public List<byte> TagValue { get; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.Tagname Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)> Tagname {get;} | |

Source code

|  |
| --- |
| public List<byte> Tagname { get; } |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTag.TlvData Property

|  |  |
| --- | --- |
| C# |  |
| public [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) TlvData {get;} | |

Source code

|  |
| --- |
| public string TlvData => StringTools.ByteArrayToHexString(TagValue.ToArray()); |

See Also

Applies to: [SmartTag](#topic_000000000000006E)

SmartTlv Class

|  |  |
| --- | --- |
| C# |  |
| public class SmartTlv | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Constructors

[SmartTlv](#topic_000000000000006C)

Properties

[TagList](#topic_000000000000006B), [TagTree](#topic_000000000000006A)

Methods

[dump](#topic_000000000000006D)

SmartTlv Constructor

|  |  |
| --- | --- |
| C# |  |
| public SmartTlv(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *tlvdata* ) | |

Parameters

tlvdata

|  |
| --- |
|  |

Source code

|  |
| --- |
| public SmartTlv(byte[] tlvdata)  {  this.tlvdata = tlvdata;  TagTree = parsetlv(tlvdata);  } |

See Also

Applies to: [SmartTlv](#topic_0000000000000069)

SmartTlv.TagList Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[SmartTag](#topic_000000000000006E)> TagList {get;} | |

Source code

|  |
| --- |
| public List<SmartTag> TagList { get; } = new List<SmartTag>(); |

See Also

Applies to: [SmartTlv](#topic_0000000000000069)

SmartTlv.TagTree Property

|  |  |
| --- | --- |
| C# |  |
| public [List](http://msdn.microsoft.com/en-us/library/6sh2ey19.aspx)<[SmartTag](#topic_000000000000006E)> TagTree {get;} | |

Source code

|  |
| --- |
| public List<SmartTag> TagTree { get; } |

See Also

Applies to: [SmartTlv](#topic_0000000000000069)

SmartTlv.dump Method

|  |  |
| --- | --- |
| C# |  |
| public [void](http://msdn.microsoft.com/en-us/library/system.void.aspx) dump(  ref [StringBuilder](http://msdn.microsoft.com/en-us/library/system.text.stringbuilder.aspx) *dumpStr*,  [int](http://msdn.microsoft.com/en-us/library/system.int32.aspx) *ident* = 0 ) | |

Parameters

dumpStr

|  |
| --- |
|  |

ident

|  |
| --- |
|  |

Source code

|  |
| --- |
| public void dump(ref StringBuilder dumpStr, int ident = 0)  {  \_dump(TagTree, ref dumpStr, ident);  } |

See Also

Applies to: [SmartTlv](#topic_0000000000000069)

StringTools Class

|  |  |
| --- | --- |
| C# |  |
| public class StringTools | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Methods

[ByteArrayToHexString](#topic_000000000000007C), [ByteArrayToHexString2](#topic_000000000000007D), [HexStringToByteArray](#topic_000000000000007E), [stringToHexString](#topic_000000000000007F)

StringTools.ByteArrayToHexString Method

|  |  |
| --- | --- |
| C# |  |
| public static [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) ByteArrayToHexString(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *bytes* ) | |

Parameters

bytes

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static string ByteArrayToHexString(byte[] bytes)  {  StringBuilder res = new StringBuilder();  return bytes.Aggregate("", (current, t) => current + t.ToString("X2"));  } |

See Also

Applies to: [StringTools](#topic_000000000000007B)

StringTools.ByteArrayToHexString2 Method

|  |  |
| --- | --- |
| C# |  |
| public static [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) ByteArrayToHexString2(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *bytes* ) | |

Parameters

bytes

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static string ByteArrayToHexString2(byte[] bytes)  {  string hex = BitConverter.ToString(bytes);  return hex.Replace("-", "");  } |

See Also

Applies to: [StringTools](#topic_000000000000007B)

StringTools.HexStringToByteArray Method

|  |  |
| --- | --- |
| C# |  |
| public static [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] HexStringToByteArray(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *hex* ) | |

Parameters

hex

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static byte[] HexStringToByteArray(string hex)  {  return Enumerable.Range(0, hex.Length)  .Where(x => x % 2 == 0)  .Select(x => Convert.ToByte(hex.Substring(x, 2), 16))  .ToArray();  } |

See Also

Applies to: [StringTools](#topic_000000000000007B)

StringTools.stringToHexString Method

|  |  |
| --- | --- |
| C# |  |
| public static [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) stringToHexString(  [string](http://msdn.microsoft.com/en-us/library/system.string.aspx) *asciistring* ) | |

Parameters

asciistring

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static string stringToHexString(string asciistring)  {  byte[] ba = Encoding.Default.GetBytes(asciistring);  var hexString = BitConverter.ToString(ba);  hexString = hexString.Replace("-", "");  return hexString;  } |

See Also

Applies to: [StringTools](#topic_000000000000007B)

TlvTools Class

|  |  |
| --- | --- |
| C# |  |
| public class TlvTools | |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

Methods

[AflParser](#topic_0000000000000088), [parseTagLengthData](#topic_0000000000000089)

TlvTools.AflParser Method

|  |  |
| --- | --- |
| C# |  |
| public static [AflResult](#topic_0000000000000080) AflParser(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *afl* ) | |

Parameters

afl

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static AflResult AflParser(byte[] afl)  {  AflResult res = new AflResult();  int i = 0;  while (i < afl.Length)  {  AflEntry entry = new AflEntry();  entry.Sfi = afl[i++] >> 3;  entry.StartRecord = afl[i++];  entry.EndRecord = afl[i++];  entry.OfflineRecords= afl[i++];  res.AflEntries.Add(entry);  }  return res;  } |

See Also

Applies to: [TlvTools](#topic_0000000000000087)

TlvTools.parseTagLengthData Method

|  |  |
| --- | --- |
| C# |  |
| public static [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] parseTagLengthData(  [byte](http://msdn.microsoft.com/en-us/library/system.byte.aspx)[] *data* ) | |

Parameters

data

|  |
| --- |
|  |

Source code

|  |
| --- |
| public static byte[] parseTagLengthData(byte[] data)  {  int index = 0;  List<byte> \_tagList = new List<byte>();  while (index < data.Length)  {  var temptag = new List<byte>();    //Get the tag name  temptag.Add(data[index]);  if ((data[index] & EmvConstants.SeeSubsequentBytes) == EmvConstants.SeeSubsequentBytes)  {  index++;  temptag.Add(data[index]);  }  Console.WriteLine("EmvTag " + StringTools.ByteArrayToHexString(temptag.ToArray()));  index++;    // Get the length of the data to follow    if ((data[index] & 0x80) == 0x80)  {  int bytesForLenght = data[index] % 0x80;  index++;  for (int i = 0; i < bytesForLenght; i++)  {  index++;  }  }  else  {  index++;  }  string tagname=StringTools.ByteArrayToHexString(temptag.ToArray()).ToUpper();  if (EmvConstants.PdolTags.ContainsKey(tagname))  {  \_tagList.AddRange(StringTools.HexStringToByteArray(EmvConstants.PdolTags[tagname]));  }  else  {  throw new Exception($"Unknown PDOL tag {tagname}");  }  }  return \_tagList.ToArray();  } |

See Also

Applies to: [TlvTools](#topic_0000000000000087)

CertificateType Enumeration

The type of the certificate

|  |  |  |
| --- | --- | --- |
| Constant | Value | Description |
| CA | 0 | Certification authority certificate |
| ICC | 1 | Issuer certificate |

Requirements

**Namespace:**[EmvLib](#topic_0000000000000001)

**Assembly:** EmvLib (in )

# Index

[AID Property](#topic_000000000000004C)

[AID Structure](#topic_0000000000000023)

[AID\_LIST Field](#topic_0000000000000022)

[AID\_LIST\_BYTES Field](#topic_0000000000000021)

[AID\_MASTERCARD\_CLASSIC Field](#topic_0000000000000024)

[AID\_VISA\_CLASSIC Field](#topic_0000000000000025)

[AID\_VISA\_ELECTRON Field](#topic_0000000000000026)

[APDU\_RES\_DESCRIPTION Field](#topic_000000000000002F)

[AflEntries Field](#topic_0000000000000081)

[AflEntry Class](#topic_0000000000000082)

[AflParser Method](#topic_0000000000000088)

[AflResult Class](#topic_0000000000000080)

[ApplicationClass Field](#topic_0000000000000016)

[Applications Property](#topic_000000000000005A)

[AuthType Property](#topic_0000000000000046)

[ByteArrayToHexString Method](#topic_000000000000007C)

[ByteArrayToHexString2 Method](#topic_000000000000007D)

[CAKey Property](#topic_0000000000000005)

[CAKeyElement Class](#topic_0000000000000002)

[CAKeys Class](#topic_0000000000000009)

[CaKey Class](#topic_000000000000000D)

[CaKeyStore Class](#topic_000000000000000B)

[CaKeys Property](#topic_000000000000000A)

[CardNotSupportedExceprion Constructor](#topic_0000000000000037)

[CardNotSupportedExceprion (String) Constructor](#topic_0000000000000038)

[CardNotSupportedExceprion (String, Exception) Constructor](#topic_0000000000000039)

[CardNotSupportedExceprion Class](#topic_0000000000000036)

[CertificateType Enumeration](#topic_000000000000003A)

[Children Property](#topic_0000000000000078)

[ConstructedDataObject Field](#topic_000000000000001A)

[ContextSpecificClass Field](#topic_0000000000000017)

[DataLen Property](#topic_0000000000000072)

[DecryptRsa (String, String) Method](#topic_0000000000000043)

[DecryptRsa (String, String, String) Method](#topic_0000000000000044)

[Description Property](#topic_0000000000000074)

[Dispose Method](#topic_0000000000000061)

[Dump Method](#topic_0000000000000058)

[EmvConstants Class](#topic_0000000000000014)

[EmvLib Namespace](#topic_0000000000000001)

[EmvLib Reference](#topic_000000000000008A)

[EmvOfflineAuthType Enumeration](#topic_0000000000000031)

[EmvRecords Property](#topic_0000000000000051)

[EndRecord Field](#topic_0000000000000085)

[Expiry Property {EmvLib.CAKeyElement}](#topic_0000000000000008)

[Expiry Property {EmvLib.CaKey}](#topic_0000000000000013)

[Exponent Property {EmvLib.CAKeyElement}](#topic_0000000000000006)

[Exponent Property {EmvLib.CaKey}](#topic_0000000000000011)

[GetAIDs Method](#topic_0000000000000060)

[GetCaKey Method](#topic_000000000000000C)

[GetOfflineTagValues Method](#topic_0000000000000057)

[GetProcessingOptions Method](#topic_0000000000000054)

[GetSha1 Method](#topic_0000000000000045)

[GetSingleApplication Method](#topic_000000000000005F)

[GetTagValue Method](#topic_0000000000000056)

[GpoTemplateFormat Enumeration](#topic_000000000000002C)

[GpoTemplateFormat Property](#topic_0000000000000052)

[HasChildren Property](#topic_0000000000000076)

[HasParent Property](#topic_0000000000000075)

[HexStringToByteArray Method](#topic_000000000000007E)

[ICC\_KEY\_HASH Field](#topic_000000000000003E)

[Index Property {EmvLib.CAKeyElement}](#topic_0000000000000004)

[Index Property {EmvLib.CaKey}](#topic_000000000000000F)

[IsConstructed Property](#topic_0000000000000079)

[IsOffline Property](#topic_0000000000000065)

[Key Property](#topic_0000000000000010)

[OfflineAuth Class](#topic_000000000000003D)

[OfflineAuth Constructor](#topic_000000000000003F)

[OfflineRecords Field](#topic_0000000000000086)

[PPSE Field](#topic_000000000000001D)

[PSE Field](#topic_000000000000001C)

[Parent Property](#topic_0000000000000077)

[PdolTags Field](#topic_0000000000000020)

[PrimitiveDataObject Field](#topic_0000000000000019)

[PrivateClass Field](#topic_0000000000000018)

[RES\_GPO Property](#topic_000000000000004E)

[RES\_READRECORD Property](#topic_0000000000000066)

[RES\_SELECT Property](#topic_000000000000004D)

[RID Property](#topic_0000000000000003)

[ReadRecords Method](#topic_0000000000000055)

[Record Property](#topic_0000000000000064)

[ResponceType Enumeration](#topic_0000000000000027)

[Rid Property](#topic_000000000000000E)

[SCardReaderExtensions Class](#topic_0000000000000047)

[SELECT\_APDU\_HEADER Field](#topic_000000000000001E)

[SHA1Hash Property {EmvLib.CAKeyElement}](#topic_0000000000000007)

[SHA1Hash Property {EmvLib.CaKey}](#topic_0000000000000012)

[SW\_SELECT\_OK Field](#topic_000000000000001F)

[SeeSubsequentBytes Field](#topic_000000000000001B)

[SelectApplication Method](#topic_0000000000000053)

[SetGpo Method](#topic_000000000000004B)

[Sfi Field](#topic_0000000000000083)

[Sfi Property](#topic_0000000000000063)

[SmartApplication Class](#topic_0000000000000049)

[SmartApplication Constructor](#topic_000000000000004A)

[SmartCard (IsoReader) Constructor](#topic_000000000000005E)

[SmartCard (SCardContext, String) Constructor](#topic_000000000000005D)

[SmartCard (String) Constructor](#topic_000000000000005C)

[SmartCard Class](#topic_0000000000000059)

[SmartEmvRecord Class](#topic_0000000000000062)

[SmartEmvRecord Constructor](#topic_0000000000000068)

[SmartTag Class](#topic_000000000000006E)

[SmartTag Constructor](#topic_000000000000007A)

[SmartTlv Class](#topic_0000000000000069)

[SmartTlv Constructor](#topic_000000000000006C)

[StartRecord Field](#topic_0000000000000084)

[StringTools Class](#topic_000000000000007B)

[TLV\_GPO Property](#topic_0000000000000050)

[TLV\_READRECORD Property](#topic_0000000000000067)

[TLV\_SELECT Property](#topic_000000000000004F)

[TagList Property](#topic_000000000000006B)

[TagStringName Property](#topic_0000000000000070)

[TagTree Property](#topic_000000000000006A)

[TagValue Property](#topic_0000000000000073)

[Tagname Property](#topic_000000000000006F)

[TlvData Property](#topic_0000000000000071)

[TlvTools Class](#topic_0000000000000087)

[TransmitWithLog Method](#topic_0000000000000048)

[UniversalClass Field](#topic_0000000000000015)

[doCdaAuth Method](#topic_0000000000000040)

[doDdaAuth Method](#topic_0000000000000041)

[doSdaAuth Method](#topic_0000000000000042)

[dump Method](#topic_000000000000006D)

[getTagDescription Method](#topic_0000000000000035)

[parseTagLengthData Method](#topic_0000000000000089)

[reader Field](#topic_000000000000005B)

[stringToHexString Method](#topic_000000000000007F)

[tags Field](#topic_0000000000000030)