

---

---

# CaptureX Dokumentation

PUBLIC

---

---



Götalandsvägen 230, hus 22,  
125 44 Älvsjö  
Sweden

Telephone +46 8 5179 5700  
Fax +46 8 5179 5707  
Internet <http://www.gemalto.com>

## Innehållsförteckning

Innehållsförteckning.....	1
Inledning.....	2
Arkitektur.....	3
CaptureX och Internet Explorer.....	4
Teknisk Specifikation.....	5
Versioner.....	5
Registrering.....	5
Klasser och metoder.....	6
OrderCollection.....	6
Order.....	9
DataAttributeCollection.....	12
DataAttribute.....	13
CustomerConfigurationRepository.....	14
CustomerConfigurationOrderBuilder.....	15
CheckedAgainstValidator.....	16
Typen.....	18

## Inledning

De personuppgifter som krävs för beställning av personaliserade kort såsom SIS-kort finns i regel redan inmatade och lagrade i något system hos kortutfärdarna. Vid beställning av ett SIS-kort via SIS Capture Station krävs att personuppgifterna matas in på nytt, vilket är både tidsödande och kan vara en potentiell felkälla.

CaptureX agerar gränssnitt mot SIS Capture Station och gör det möjligt för externa programkomponenter att i förväg behandla och spara beställningsinformation. Beställningar införda via CaptureX kompletteras i SIS Capture Station (SCS) varpå de signeras, krypteras och skickas till Gemalto.

Det är möjligt att via CaptureX skapa och behandla beställningar gällande både SIS-kort och icke SIS-kort. Vilken korttyp man avser behandla anges när man skapat en beställning och SIS Capture Station agerar på olika sätt beroende på vilken korttyp det rör sig om.

## Arkitektur

CaptureX består av både COM-komponenter och .NET assemblies. Klasserna som exponeras via COM är:

- OrderCollection – En samlingsklass för inmatade beställningar
- Order – En klass för att representera den information som beställning av ett kort kan innehålla.
- DataAttributeCollection – En samlingsklass för attribut till de data som finns i Order.
- DataAttribute – En klass för att representera attribut till den information som beställning av ett kort kan innehålla.
- CustomerConfigurationRepository – En klass som används för att läsa ut information om de nedladdade kunduppgifterna.
- LocalConfigurationOrderBuilder – En klass som används för att skapa beställningar utifrån den lokala konfigurationsfilen.
- CustomerConfigurationOrderBuilder – En klass som används för att skapa beställningar utifrån de nedladdade kunduppgifterna samt den lokala konfigurationsfilen.
- CheckedAgainstValidator – En klass som används för att validera fälten för personnummerkontroll.

För att använda sig av CaptureX börjar man lämpligen med att skapa en instans av OrderCollection. Datasamlingen OrderCollection innehåller alltid de beställningar som finns inlagda via CaptureX sedan tidigare men som ännu inte har behandlats i SIS Capture Station.

Från och med Windows Vista och Internet Explorer 7 finns en ny säkerhetsfunktion i webbläsaren som kallas ”skyddat läge”. Denna funktion kan göra att CaptureX inte fungerar som tänkt ifrån webbplatser som inte uttryckligen markerats som betrodda platser i Internet Explorer. Får du upp en varningsdialog om skyddat läge när du skapar en instans av OrderCollection körs din webbläsare i ”skyddat läge”, och du behöver lägga till den webbplats som använder sig av CaptureX bland din webbläsares betrodda platser. Mer information hittar du här: [Säkerhetszoner: lägga till eller ta bort webbplatser](#).

För att skapa en ny beställning utgår man från en ny instans av klassen Order. De attribut som klassen Order har motsvarar de fält som finns vid inmatning av en beställning i SCS (Forenames, Surname, etc). Det görs ingen kontroll att den information som matas in i klassen Order är korrekt, kontroll av vissa kritiska fält såsom personnumrets checksiffra sker dock i SCS.

Vill man istället skapa en beställning utifrån de kunduppgifter som laddats ner kan man använda klassen CustomerConfigurationOrderBuilder. Då anger man först tumavtrycket på beställarens certifikat, och sedan skapar man beställningen utifrån kund- och produktkoden. LocalConfigurationOrderBuilder kan användas ifall man inte har tillgång till nedladdade kunduppgifter, utan bara SIS Capture Stations egna konfiguration.

De beställningar som förs in i CaptureX sparas på disk i en XML-fil, där de senare kan läsas in i SIS Capture Station. När CaptureX avslutas krypteras filen med beställningsinnehållet. Varje förändring i någon av Order-klasserna i samlingen leder till att XML-filen uppdateras.

Vill man hantera personnummerkontrollen utanför SIS Capture Station, men fortfarande vill kunna kontrollera giltigheten på körkorts- och SIS-kortnummer kan man använda sig av klassen CheckedAgainstValidator för att kontrollera dessa.

## **CaptureX och Internet Explorer**

Klasserna OrderCollection, Order, CustomerConfigurationRepository, LocalConfigurationOrderBuilder, CustomerConfigurationOrderBuilder och CheckedAgainstValidator implementerar samtliga gränssnittet IObjectSafety vilket gör att vissa varningsmeddelanden kan undvikas när komponenten körs i Microsoft Internet Explorer.

CaptureX har testats i Internet Explorer version 6 till 11.

## **Teknisk Specifikation**

### ***Versioner***

CaptureX är implementerat i .NET 4 Client Profile och C#.

### ***Registrering***

CaptureX.dll och Gemalto.SISCaptureStation.Common.dll måste bägge registreras som ActiveX-kontroller samt installeras i Global Assembly Cache. Även Gemalto.SISCaptureStation.ConfigurationLib.dll, samt Gemalto.SisCaptureStation.Shared.dll som de bägge andra beror på måste installeras i Global Assembly Cache.

CaptureX använder SCSs nycklar i registret på plats HKLM/SOFTWARE/Gemalto/SIS Capture Station.

Registrering av dll-filer och registernycklar sker i SIS Capture Stations installationsprogram.

## Klasser och metoder

### **OrderCollection**

Implementerar IDisposable, IEnumerable, IObjectSafety

<code>OrderCollection();</code>	
Beskrivning	Öppnar samlingsobjektet för beställningar  Varje förändring som sker i <code>OrderCollection</code> eller med någon av de beställningar som finns lagrade i den återspeglas i en XML-fil
Undantag	<code>ApplicationException</code> – Om en annan applikation håller låset till CaptureX:s XML-fil

<code>CaptureXOrder Add(CaptureXOrder newEntry);</code>	
Beskrivning	Lägger till en ny beställning till samlingen
Parametrar	<code>newEntry</code> – Den nya beställningen
Returnerar	Den tillagda beställningen
Undantag	<code>ArgumentException</code> – Beställningen finns redan tillagd i samlingen

<code>void Clear();</code>	
Beskrivning	Rensar samlingen på beställningar

<code>bool Contains(CaptureXOrder order);</code>	
Beskrivning	Kontrollerar om en beställning är tillagd i samlingen
Parametrar	<code>order</code> – Beställningen att söka efter
Returnerar	Sant om beställningen fanns i samlingen

<code>int Count();</code>	
Beskrivning	Räknar antalet beställningar i samlingen
Returnerar	Antalet beställningar i samlingen

<code>void Dispose();</code>	
Beskrivning	Stänger ner objektet och frigör minne och övriga resurser

<code>CaptureXOrder GetByOrderId(string orderId);</code>	
Beskrivning	Används för att hämta en beställning ur samlingen givet dess id
Parametrar	<code>orderId</code> – Beställningens Id
Returnerar	Beställningen med det angivna id:t, eller <code>null</code> om beställningen inte kunde hittas i samlingen

<code>CaptureXOrder Item(int index);</code>	
Beskrivning	Hämtar en beställning med givet index ur samlingen
Parametrar	<code>index</code> – Beställningens index i samlingen
Returnerar	Beställningen med det givna indexet
Undantag	<code>IndexOutOfRangeException</code> – Om index är utanför samlingen.

<code>void LaunchSCS(string LaunchParams);</code>	
Beskrivning	Startar SCS för komplettering och signering av beställning

Parametrar	LaunchParams – Parametrar som anger hur i vilket läge SIS Capture Station ska startas
Kommentar	<p>När metoden anropas släpper CaptureX skrivrättigheter till lagringsfilen. Möjliga värden för launchParams är:</p> <p>Ett av följande startalternativ: /continue – Startar SCS i läget för att behandla befintliga beställningar /new – Startar SCS i läget för att skapa en ny beställning /standard – Startar SCS som vanligt /orderId: [Order.OrderId] – Startar SCS i läget för att behandla den angivna beställningen</p> <p>Ett av följande avslutsalternativ: /exit:allowMulti – Tillåt att flera beställningar behandlas /exit:allowOne – Avsluta efter att en beställning behandlats</p> <p>Ett av följande alternativ för kortläge: /card: [SIS !SIS] – Anger huruvida SIS eller icke-SIS kort ska behandlas. Default är SIS. Om flaggan /orderId är satt kommer läget SCS startas i bero på den angivna orderns typ</p>
Exempel	<p>Om metoden anropas med: LaunchSCS ("/new /exit:allowOne /card:!SIS")</p> <p>kommer SCS startas med dialogfönstret för ett nytt icke SIS-kort. SCS kommer att avslutas om orderinmatningen avbryts eller om ordern signeras, CaptureX kommer då återfå kontrollen och ladda XML-filen på nytt</p>
Undantag	FileNotFoundException – SCS kunde inte hittas

```
static void LaunchSCS(ExitPolicy policy, ActionAfterSCSRun action, CaptureXOrder order);
```

Beskrivning	Startar SCS för komplettering och signering av beställning
Parametrar	<p>policy – Ett värde ur ExitPolicy</p> <p>action – Ett värde ur ActionAfterSCSRun</p> <p>order – Ordern som SCS ska startas med</p>
Undantag	FileNotFoundException – SCS kunde inte hittas

```
void LaunchSCS(ExitPolicy policy, ActionAfterSCSRun action, LaunchMode mode, CardType type);
```

Beskrivning	Startar SCS för komplettering och signering av beställning
Parametrar	<p>policy – Ett värde ur ExitPolicy</p> <p>action – Ett värde ur ActionAfterSCSRun</p> <p>mode – Ett värde ur LaunchMode</p> <p>type – Ett värde ur CardType</p>
Undantag	FileNotFoundException – SCS kunde inte hittas

```
string LaunchSCSWithCertThumbprint { get; set; }
```

Beskrivning	Anger vilket certifikat SIS Capture Station ska använda sig av när det startas från CaptureX
Värde	Tumavtrycket för det certifikat SCS skall använda sig av



Exempel	<pre>OrderCollection coll; X509Certificate2 cert;  coll.LaunchSCSWithCertThumbprint = cert.Thumbprint;</pre>
---------	--

<code>CaptureXOrder Remove(int index);</code>	
Beskrivning	Tar bort en beställning med ett givet index ur samlingen
Parametrar	index – Beställningens index i samlingen
Returnerar	Den borttagna beställningen
Undantag	<code>IndexOutOfRangeException</code> – Angivet index finns inte i samlingen

<code>void RemoveOrder(CaptureXOrder order);</code>	
Beskrivning	Tar bort en given beställning ur samlingen
Parametrar	order – Den beställning som ska tas bort ur samlingen
Undantag	<code>ArgumentException</code> – Angiven beställning finns inte i samlingen

<code>string Version { get; }</code>	
Beskrivning	Returnerar CaptureX versionsnummer
Värde	CaptureX Versionsnummer

<code>bool IsRunningInProtectedMode();</code>	
Beskrivning	Kontrollerar ifall objektet har skapats från en Internet Explorer i skyddat läge
Returnerar	Sant ifall objektet har skapats från en Internet Explorer i skyddat läge

## Order

<code>Order(OrderType type, StorageHandler storage);</code>	
Beskrivning	<p>Skapar en ny tom beställning med de angivna parametrarna</p> <p>De attribut klassen innehåller motsvaras av den information som matas in i SCS</p> <p>En förändring i en beställning återspeglas i den sparade XML-filen i det fall beställningen är en medlem av samlingsklassen OrderCollection</p>
Parametrar	<p><code>type</code> – Ett värde ur <code>OrderType</code></p> <p><code>storage</code> – Ett värde ur <code>StorageHandler</code></p>
Kommentar	Prestandamässigt är det att föredra att fylla en beställning med data innan det läggs till i samlingen

<code>Order(OrderType type, string xml, string orderid, bool headerOnly, StorageHandler storage);</code>	
Beskrivning	<p>Öppnar en befintlig beställning</p> <p>De attribut klassen innehåller motsvaras av den information som matas in i SCS</p> <p>En förändring i en beställning återspeglas i den sparade XML-filen i det fall beställningen är en medlem av samlingsklassen OrderCollection</p>
Parametrar	<p><code>type</code> – Ett värde ur <code>OrderType</code></p> <p><code>xml</code> – Den befintliga beställningens XML-data</p> <p><code>orderid</code> – Den befintliga beställningens id</p> <p><code>headerOnly</code> – Ifall hela beställningen skall laddas om från fil</p> <p><code>storage</code> – Ett värde ur <code>StorageHandler</code></p>
Kommentar	Prestandamässigt är det att föredra att fylla en beställning med data innan det läggs till i samlingen

<code>bool SetPhoto(string picturePath);</code>	
Beskrivning	Anger ett fotografi till beställningen
Parametrar	<code>picturePath</code> – Sökväg till bilden som ska användas
Returnerar	Sant om bilden kunde användas

<code>bool SetPhoto(Bitmap picture);</code>	
Beskrivning	Anger ett fotografi till beställningen
Parametrar	<code>picture</code> – Bilden som ska användas
Returnerar	Sant om bilden kunde användas

<code>bool SetPhoto(byte[] pictureBytes);</code>	
Beskrivning	Anger ett fotografi till beställningen
Parametrar	<code>pictureBytes</code> – Den serialiserade bilden som ska användas
Returnerar	Sant om bilden kunde användas

<code>byte[] GetPhotoBytes();</code>	
Beskrivning	Hämtar beställningens fotografi i serialiserad form

Returnerar	Beställningens fotografi i serialiserad form
------------	--

```
Bitmap GetPhoto();
```

Beskrivning	Hämtar beställningens fotografi
Returnerar	Beställningens fotografi

```
bool SetSignPicture(string strSignPicPath);
```

Beskrivning	Anger en signatur till beställningen
Parametrar	strSignPicPath – Sökväg till bilden som ska användas
Returnerar	Sant om bilden kunde användas

```
bool SetSignPicture(Bitmap SignPicture);
```

Beskrivning	Anger en signatur till beställningen
Parametrar	SignPicture – Bilden som ska användas
Returnerar	Sant om bilden kunde användas

```
bool SetSignPicture(byte[] bSignPicture);
```

Beskrivning	Anger en signatur till beställningen
Parametrar	bSignPicture – Den serialiserade bilden som ska användas
Returnerar	Sant om bilden kunde användas

```
byte[] GetSignPicBytes();
```

Beskrivning	Hämtar beställningens signatur i serialiserad form
Returnerar	Beställningens signatur i serialiserad form

```
public Bitmap GetSignPic();
```

Beskrivning	Hämtar beställningens signatur
Returnerar	Beställningens signatur

```
string strOrderId { get; }
```

Beskrivning	Returnerar beställningens id
Returnerar	Beställningens id

Attribut	Vid hantering av en beställning kan en valfri mängd av beställningens attribut sättas. Attributen motsvarar den information som kan finnas med i en kortbeställning:
	<pre> string AdditionalInfo { get; set; } string Address { get; set; } string BarcodeType { get; set; } int CardType { get; set; } string CardTypeString { get; } string CheckedAgainst { get; set; } string CheckedAgainstValue { get; set; } string City { get; set; } string Customer { get; set; } string CustomField { get; set; } string CustomerOrderId { get; set; } string Email { get; set; } EyeColor EyeColor { get; set; } string EyeColorString { get; set; } string Forenames { get; set; } string GivenNameCode { get; set; } HairColor HairColor { get; set; } string HairColorString { get; set; } string Height { get; set; } string MagStripe1 { get; set; } string MagStripe2 { get; set; } string MagStripe3 { get; set; } string Office { get; set; } string Organization { get; set; } string OrgUnit { get; set; } string PersonalID { get; set; } string PostalCode { get; set; } string Product { get; set; } bool ProtectedId { get; set; } string Surname { get; set; } string UserID { get; set; }  DataAttributeCollection DataAttributes { get; } </pre>

## **DataAttributeCollection**

`DataAttributeCollection()` ;

Beskrivning	Skapar ett nytt samlingsobjekt för attributinformation
-------------	--

`DataAttribute` `GetDataAttribute(string Key)` ;

Beskrivning	Hämtar attributinformation för det angivna attributet
Parametrar	<code>Key</code> – Anger för vilket attribut som information ska hämtas
Returnerar	Attributinformationen för det angivna attributet
Undantag	<code>ArgumentException</code> – Ett okänt attribut angavs

`DataAttribute` `GetDataAttribute(DataAttributeNamesEnum Key)` ;

Beskrivning	Hämtar attributinformation för det angivna attributet
Parametrar	<code>Key</code> – Anger för vilket attribut som information ska hämtas
Returnerar	Attributinformationen för det angivna attributet

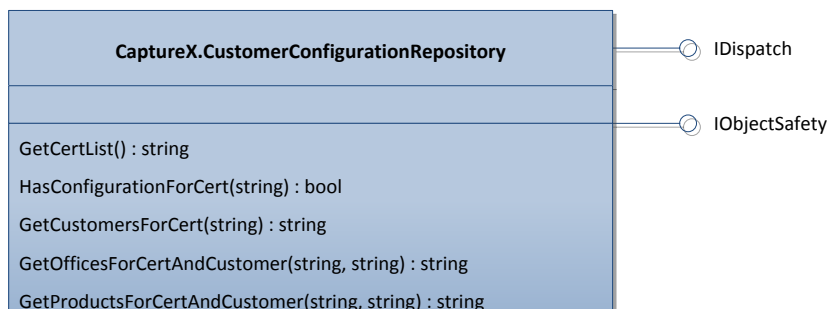
## DataAttribute

DataAttribute();

Beskrivning Skapar ett nytt attributinformatjonsobjekt

Attribut	Vid hantering av ett attributinformatjonsobjekt kan valfri mängd av objektets attribut hanteras
	<p><code>string Label { get; set; }</code> – Sätter eller returnerar denna beställnings rubrik för detta attributinformatjonsobjekt. <code>null</code> innebär att man använder inställningarna från konfigurationen</p> <p><code>FieldStatusEnum ReadOnly { get; set; }</code> – Sätter eller returnerar om detta attribut ska vara skrivskyddat</p> <p><code>string ReadOnlyString { get; set; }</code> – Sätter eller returnerar om detta attribut ska vara skrivskyddat. Värdet som sätts tolkas som <code>FieldStatusEnum</code></p> <p><code>FieldStatusEnum Visible { get; set; }</code> – Sätter eller returnerar om detta attribut ska vara synligt</p> <p><code>string VisibleString { get; set; }</code> – Sätter eller returnerar om detta attribut ska vara synligt. Värdet som sätts tolkas som en <code>FieldStatusEnum</code></p>

## CustomerConfigurationRepository



Klassen CustomerConfigurationRepository används för att läsa ut information om de nedladdade kunduppgifterna. Denna information kan användas för att säkerställa att kunduppgifterna verkligen laddats ner och är korrekta för den aktuella användaren, innan en ny order skapas med hjälp av klassen CustomerConfigurationOrderBuilder.

<code>string GetCertList()</code>	
Beskrivning	Hämtar de certifikat det finns nedladdade kunduppgifter för.
Parametrar	
Returnerar	En semikolon-separerad lista av vilka certifikat det finns nedladdade kunduppgifter för.
Undantag	

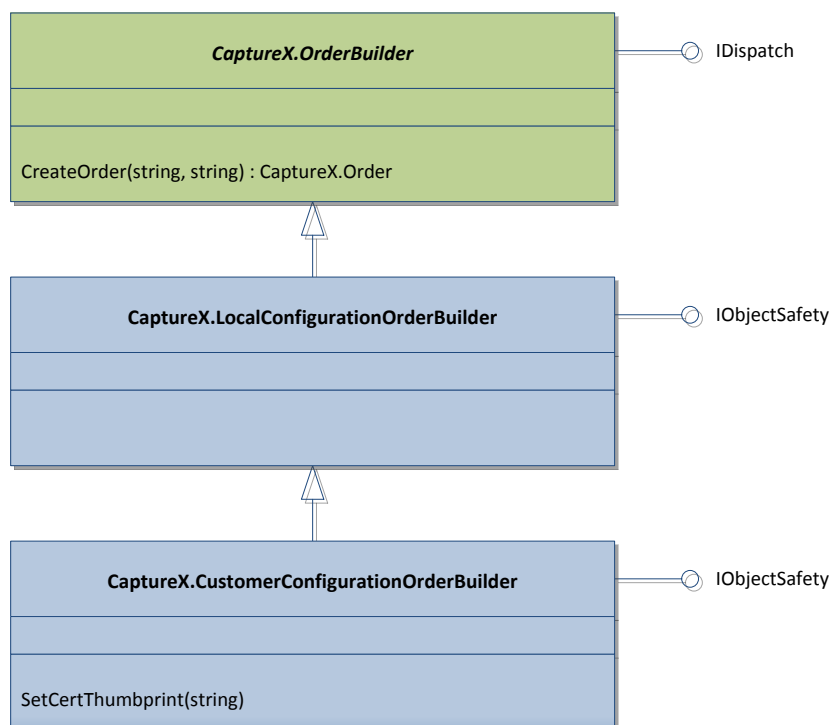
<code>bool HasConfigurationForCert(string certThumbprint)</code>	
Beskrivning	Kontrollerar ifall kunduppgifter har laddats ner för det angivna certifikatet.
Parametrar	<code>certThumbprint</code> – Tumavtrycket på certifikatet att kontrollera.
Returnerar	Sant om det finns kunduppgifter nedladdade för certifikatet.
Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som tumavtryck.

<code>string GetCustomersForCert(string certThumbprint)</code>	
Beskrivning	Hämtar kunder för det angivna certifikatet.
Parametrar	<code>certThumbprint</code> – Tumavtrycket på certifikatet att hämta kunder för.
Returnerar	En semikolon-separerad lista av kunder som certifikatets ägare är behörig att beställa för.
Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som tumavtryck. <code>KeyNotFoundException</code> – Ett okänt tumavtryck angavs.

<code>string GetOfficesForCertAndCustomer(string certThumbprint, string customerCode)</code>	
Beskrivning	Hämtar kontor för det angivna certifikatet och kunden.
Parametrar	<code>certThumbprint</code> – Tumavtrycket på certifikatet att hämta kontor för. <code>customerCode</code> – Kundkoden att hämta kontor för.
Returnerar	En semikolon-separerad lista av kontor som certifikatets ägare är behörig att beställa för den angivna kunden.
Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som tumavtryck. <code>KeyNotFoundException</code> – Ett okänt tumavtryck eller kund angavs.

<code>string</code> GetProductsForCertAndCustomer( <code>string</code> certThumbprint, <code>string</code> customerCode)	
Beskrivning	Hämtar produkter för det angivna certifikatet och kunden.
Parametrar	certThumbprint – Tumavtrycket på certifikatet att hämta produkter för. customerCode – Kundkoden att hämta produkter för.
Returnerar	En semikolon-separerad lista av produkter som certifikatets ägare är behörig att beställa för den angivna kunden.
Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som tumavtryck. <code>KeyNotFoundException</code> – Ett okänt tumavtryck eller kund angavs.

## CustomerConfigurationOrderBuilder



Klassen CustomerConfigurationOrderBuilder används för att skapa beställningar utifrån de nedladdade kunduppgifterna samt den lokala konfigurationsfilen. Den sätter alla de värden och övriga attribut på fälten i beställningen efter vad som angivits i konfigurationen och i kunduppgifterna.

Klassen CustomerConfigurationOrderBuilder ärver av klassen LocalConfigurationOrderBuilder som kan användas som komplement i de fall man inte har tillgång till nedladdade kunduppgifter. Beställningen skapas i så fall bara utifrån vad som angivits i konfigurationen, utan att ta hänsyn till eventuella kunduppgifter.

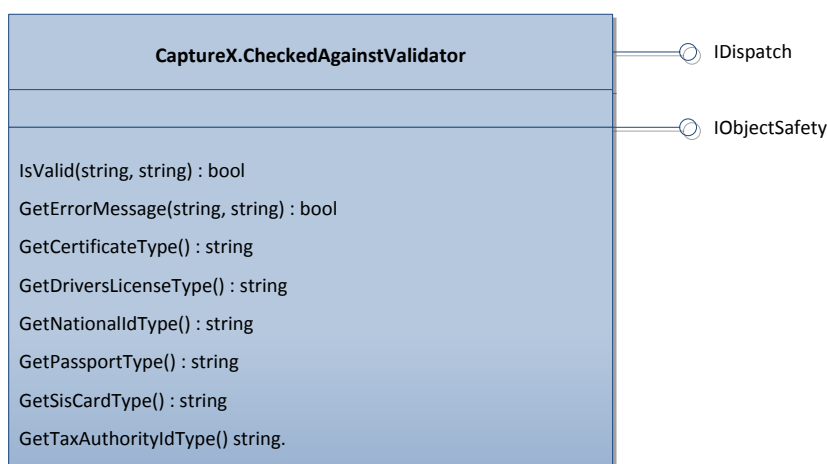
<code>SetCertThumbprint</code> ( <code>string</code> certThumbprint)	
Beskrivning	Anger vilket certifikat som skall användas för att slå upp kunduppgifterna som beställningen skall skapas utifrån.
Parametrar	certThumbprint – Tumavtrycket på certifikatet att skapa beställningen ifrån.
Returnerar	



Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som tumavtryck. <code>KeyNotFoundException</code> – Ett okänt tumavtryck angavs.
----------	---

<code>Order CreateOrder(string customerCode, string productCode)</code>	
Beskrivning	Skapar en ny beställning utifrån innehållet i de nedladdade kunduppgifterna samt den lokala konfigurationsfilen.
Parametrar	<code>customerCode</code> – Kundkoden att slå upp kunduppgifterna för. <code>productCode</code> – Produktkoden att slå upp kunduppgifterna för.
Returnerar	Den skapade beställningen
Undantag	<code>ArgumentNullException</code> – En nollreferens angavs som kund eller produkt. <code>InvalidOperationException</code> – Certifikatets tumavtryck har inte angivits. <code>ArgumentException</code> – Certifikatets ägare är inte behörig till antingen kunden eller produkten.

## CheckedAgainstValidator



Klassen `CheckedAgainstValidator` används för att validera de värden som angivits för personnummerkontrollen, ifall detta skall skötas utanför SIS Capture Station. Den kod SCS använder internt för kontrollen exponeras genom denna klass.

För att köra en validering används metoden `IsValid`, som tar två strängparametrar: typen av personnummerkontroll som använts, och så själva värdet. För att få de typer som stöds kan hjälpmetoderna `GetCertificateType`, `GetDriversLicenseType`, `GetNationalIdType`, `GetPassportType`, `GetSisCardType` och `GetTaxAuthorityIdType` användas. Dessa motsvarar då identifieringstyperna intygsnummer, körkortnummer, nationellt ID-kortnummer, passnummer, SIS-kortnummer och Skatteverkets ID-kortnummer.

Misslyckas valideringen kan man läsa ut ett felmeddelande med hjälp av metoden `GetErrorMessage`. Den tar samma inparametrar som `IsValid`.

NB: Klienten accepterar idag beställningar även ifall SIS-kortnummervalidering misslyckas, på grund av att så kallade ”stående kort” med SIS-nummer inte visar hela kortnumret. I annat fall hade dessa kort inte kunnat användas för personnummerkontroll.

<code>IsValid(string type, string value)</code>	
Beskrivning	Validerar de värden som angivits för personnummerkontrollen

Parametrar	<p><code>type</code> – En av de kontrolltyper som stöds. Stödda typer kan hämtas med hjälpmetoderna <code>GetCertificateType</code>, <code>GetDriversLicenseType</code>, <code>GetNationalIdType</code>, <code>GetPassportType</code>, <code>GetSisCardType</code> och <code>GetTaxAuthorityIdType</code>.</p> <p><code>value</code> – Värdet på personnummerkontrollen – alltså själva passnumret, intygsnumret etc. som ligger till grund för kontrollen.</p>
Returnerar	Sant om valideringen lyckades

<code>GetErrorMessage(string type, string value)</code>	
Beskrivning	Hämtar ett eventuellt felmeddelande vid misslyckad validering.
Parametrar	<p><code>type</code> – En av de kontrolltyper som stöds. Se <code>IsValid</code> ovan för ytterligare information.</p> <p><code>value</code> – Värdet på personnummerkontrollen – alltså själva passnumret, intygsnumret etc. som ligger till grund för kontrollen.</p>
Returnerar	Aktuellt felmeddelande om valideringen misslyckades, eller den tomma strängen ifall valideringen lyckades.

<code>GetCertificateType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen intygsnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "Certificate".

<code>GetDriversLicenseType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen körkortsnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "Drivers license".

<code>GetNationalIdType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen nationellt ID-kortnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "National ID".

<code>GetPassportType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen passnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "Passport".

<code>GetSisCardType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen SIS-kortnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "SIS".

<code>GetTaxAuthorityIdType()</code>	
Beskrivning	Hämtar namnet på kontrolltypen Skatteverkets ID-kortnummer.
Parametrar	
Returnerar	Namnet på kontrolltypen intygsnummer: "Tax Authority ID".

## Typer

enum DataAttributeNamesEnum;	
Beskrivning	Används när ett attribut ska hämtas från en beställning
Möjliga värden	AdditionalInfo, Address, BarCodeType, CardType, City, Civilservant, Customer, CustomerOrderId, Customfield, Email, Forenames, GivenNameCode, MagStripe1, MagStripe2, MagStripe3, Office, Organization, OrganizationalUnit, PersonalId, Photo, PostalCode, Product, SignPicture, Surname, UserId, Verified

enum FieldStatusEnum;	
Beskrivning	Används när man definierar om ett attribut ska vara synligt eller skrivskyddat
Möjliga värden	Undefined – Ingen specifik inställning för attributet, utan det är konfigurationen som styr True – Anger att attributet är skrivskyddat eller synligt False – Anger att attributet inte är skrivskyddat eller synligt

enum LaunchMode;	
Beskrivning	Används när SCS startas från CaptureX. Anger vilket läge programmet ska startas i
Möjliga värden	Standard – Startar SCS i standardläge ContinueCardOrder – Startar i läget för behandling av befintliga beställningar NewCardOrder – Startar i läget för skapande av nya beställningar SendOrdersToSetec – Startar i läget för att skicka in beställningar

enum CardType;	
Beskrivning	Anger vilken typ av kort som ska behandlas
Möjliga värden	SISCard – Behandla SIS-kort NonSISCard – Behandla icke SIS-kort

enum ExitPolicy;	
------------------	--

Beskrivning	Anger hur CaptureX ska bete sig när användaren avbryter eller signerar en kortbeställning
Möjliga värden	<p>AllowMultipleOrders– Tillåt att flera beställningar behandlas, signeras samt skickas till Gemalto</p> <p>ExitOnSign– Tillåt endast behandling av en beställning. Om användaren avbryter eller signerar beställningen avslutas SCS</p>

`enum HairColor;`

Beskrivning	Anger hårfärg i en beställning
Möjliga värden	<p>EjSpecificerat, Svart, Brunt, Mörkblont, Mellanblont, Ljusblont, Gråsprängt, Grått, Vitt, Rött, Skallig</p>
Kommentar	<p>Vill man hellre ange hårfärg som sträng finns attributet <code>HairColorByString</code>. Tillåtna värden är de från enumerationen ovan</p>

`enum EyeColor;`

Beskrivning	Anger ögonfärg i en beställning
Möjliga värden	<p>EjSpecificerat, Mörkbruna, Ljusbruna, Gulaktiga, Blå, Grå, Gröna, Melerade, Olikfärgade</p>
Kommentar	<p>Vill man hellre ange ögonfärg som sträng finns attributet <code>EyeColorByString</code>. Tillåtna värden är de från enumerationen ovan</p>