

התוכנה הפשוטה והמשתלמת להנהלת חשבונות



ראו כמה זה פשוט וקל ✓
חסכון זמן ושעות תיוק ✓

התוכנה המתקדמת לבעלי עסקים

ה-API של כספית CaspitWeb-ו

מחבר: דרור סדן dror@caspit.biz

עבור API גרסה: 3.2.0.95 ומעלה

ה-API מתאים לכספית גרסה: 3.293 ומעלה

עדכון אחרון: 24/5/2012

כספית תוכנה בע"מ

ת.ד. 164 • עין שריד • 40697

www.caspit.biz

טל. 077-9233325 • פקס 077-9233326

תוכן העניינים

תוכן העניינים.....	2
הקדמה	6
-API למה מתאים ה-.....	6
API מבוסס COM	7
-API דרישות מוקדמות להפעלה מוצלחת של ה-.....	7
-API מה הטכנולוגיה של ה-.....	7
-API כיצד מתחברים ל-.....	7
-API הפעלה ב-XML	8
-API הפעלה של ה- DebugMode	8
משלוח מייל	9
שליטה במספר העותקים המודפסים (3.2).....	9
גרסה API 3.2 של כספית 3.1 ל-API הבדלים בין ה-.....	11
של רשומות בכספית ID3.2-ה.....	11
שינויים נוספים	11
בגרסה API 3.2 סימון של חידושים ב-.....	12
-API מפת כיסוי של ה-.....	13
Properties תכונות.....	15
לקוחות.....	16
- יצירה של לקוח CreateCustomer.....	16
XML לפי - יצירה של לקוח CreateCustomerXML.....	16
- שליפה של לקוח לפי מפתח זר GetCustomerByForeignKey.....	17
- שליפה של לקוח לפי מפתח זר GetCustomerByForeignKeyXML.....	17
- שליפה של יתרת הלקוח לפי המפתח בכספית GetCustomerBalanceById.....	17
- שליפה של יתרת הלקוח לפי מפתח זר GetCustomerBalanceByForeignKey.....	17
- שליפה של כל הלקוחות של כספית GetCustomers.....	17
- שליפה של כל הלקוחות של כספית GetCustomersXML.....	17
- אובייקט לקוח Customer.....	17
-- עדכון של לקוח לפי מפתח זר UpdateCustomerByForeignKey.....	18
-- עדכון של לקוח לפי מפתח זר UpdateCustomerByForeignKeyXML.....	19
- שליפה של יתרת לקוחות GetCustomersBalance.....	19
דוגמת קוד לשליפת יתרת לקוחות.....	20
- עדכון של יתרת הלקוחות RefreshCustomerBalance(3.2).....	20
אובייקטים משניים.....	22
TrxCode סיווג תנועה -	22
GetTrxCodes שליפה של כל סיווגי התנועות -	22

– שליפה של כל סיווגי התנועות – GetTrxCodesXML(3.2)	22
שלומו PaymentTerms תנאי	22
חשבוניות	24
חשבונית CaspitAPI.Invoice אובייקט	24
– CaspitAPI.LineItem שורת חשבונית	25
– CreateInvoice יצירה של חשבונית	26
XML – CreateInvoiceXML על-ידי יצירה של חשבונית	26
שליפה של חשבונית לפי מספר החשבונית	26
– GetInvoices שליפה של כל החשבוניות שיש בכספית	27
-- הצגה של חשבונית DisplayInvoice	27
– הדפסה של חשבונית PrintInvoice	28
קבלות	30
– CreateReceipt יצירה של קבלה	30
– ReceiptLineItem שורת קבלה	30
XML – CreateReceiptXML יצירה של קבלה על-ידי	31
הדפסה של קבלה	31
חשבונית/קבלה	32
יצירה של חש/קבלה	32
XML יצירה של חש/קבלה ב-	32
הצגה של חש/קבלה	32
הדפסה של חש/קבלה	32
שליפה של חש/קבלה	32
שליפה של כל החש/קבלה במערך	32
חשבונית עסקה	33
יצירה של חשבונית עסקה	33
הצגה של חשבונית עסקה בכספית	33
הדפסה של חשבונית עסקה	33
חשבונית זיכוי	34
יצירה של חשבונית זיכוי	34
XML יצירה של חשבונית זיכוי ב-	34
הצגה של חשבונית זכוי בכספית	34
הדפסה של חשבונית זכוי	34
תעודות החזר	35
– CreateRefund יצירה של תעודת החזר	35
XML – CreateRefundXML יצירה של תעודת החזר על-ידי	35
הצגה של תעודת החזר בכספית	35
הדפסה של תעודת החזר	35

חיובי זמן.....	36
'יצירת חיוב זמן.....	36
מחירונים.....	37
פריטים.....	38
פעולות נוספות.....	39
הרצה של כספית.....	39
שליפה של אחוז המע"מ.....	39
של כספיתRegistryשליפה של ערך.....	39
שליפה של ערך מטבלת הפרמטרים של כספית.....	39
שינוי של המדפסת של כספית.....	39
משלוח של מסמך בדואר אלקטרוני.....	39
משלוח מייל.....	40
ביצוע גיבוי.....	40
הוסף הודעת שגיאה לטבלת שגיאות של כספית.....	41
XMLתיאור של ה-.....	42
NULLעדכון ערכים, שדות ריקים ו-.....	42
עבור יצירת לקוח ועדכוןXML.....	43
בסיסי עבור חשבוניות מס, חשבוניות עסקה, חשבוניות זיכויXML.....	43
ערכים.....	44
מטבעות.....	44
דוגמא – יצירת חשבונית.....	44
עבור יצירת קבלהXML.....	45
עבור יצירת תעודת החזרXML.....	46
דוגמא – יצירת קבלה.....	47
עבור שליפת חשבוניתXML.....	48
עבור רשימת לקוחותXML.....	49
עבור יצירת חשבונית מס/קבלהXML.....	50
עבור יצירת חשבונית זיכויXML.....	51
עבור סיווגי תנועהXML.....	51
דוגמאות קוד.....	53
דוגמה ליצירת לקוח.....	53
דוגמה ליצירת חשבונית.....	53
דוגמה להדפסת חשבונית באנגלית.....	54
Registry וה-API.....	55
Web API.....	56
APIדרישות מוקדמות להפעלה מוצלחת של ה-.....	56
API?מה הטכנולוגיה של ה-.....	56

-API כיצד מתחברים ל-.....	56
יצירת חשבונית.....	56
יצירת חש/קבלה.....	56
שליפת לקוחות.....	56
שליפת חשבוניות.....	56
שליפת חש/קבלה.....	56
נספחים.....	57
נספח א – רשימת בנקים.....	57
נספח ב' – רשימת סוגי תשלום.....	57
נספח ג' – סטטוסים של מסמכים שנשלפים מכספית.....	58
נספח ד' – סוגי מסמכים בכספית.....	58
נספח ה' – רשימה של סוגי תנועה שמוגדרים בכספית.....	58
האם המשתמש יכול לשנות את המספר של סיווג התנועה?.....	60

הקדמה

חברת כספית תוכנה, המפתחת והמשווקת של כספית - התוכנה הקלה והמשתלמת להנהלת חשבונות, משווקת את כספית יחד עם API. לכספית שני סוגים של API:

1. מבוסס COM (כלומר DLL המותקן על המחשב של המשתמש) המאפשר להפעיל את כספית כמודול הנהלת חשבונות בתוכנות שולחניות אחרות.

2. מבוסס Web המאפשר להפעיל את כספית מאתר אינטרנט (נמצא בכספית 3.293 והילך)

המסמך מתאר את שני הסוגים של ה-API של תוכנת כספית, עם דגשים על הבדלים בין גרסת כספית 3.1 לגרסת כספית 3.2.

למה מתאים ה-API?

הנה טבלה שממפה את היכולות של ה-API ולמה הוא מתאים:

מתאים ל...	
לכל תוכנה שולחנית (Desktop) מבוססת חלונות, שדרוש לה מודול הנהלת חשבונות. הדרישה היחידה היא יכולה להפעיל ActiveX. כל פלטפורמות הפיתוח מ-10 השנים האחרונות מאפשרות זאת. דוגמאות לשילוב של כספית בתוכנות אחרות שפותחו בטכנולוגיות שונות: ■ מערכת לחיוב לקוחות בחברת מסחר בבורסה. המערכת פותחה ב-C# (.NET). ■ מערכת לניהול חוגי פילאטיס. המערכת פותחה ב-C++. מערכת לניהול סקרי בטיחות של מעליות. המערכת פותחה באקס.ס.	API מבוסס COM
אתרי אינטרנט שרוצים להפיק חשבונות מס (או חש/קבלה) בסיום תהליך הרכישה של הלקוח. אפליקציות Web כמו CRM הרוצים להעביר את הטיפול בלקוח לאחר ביצוע הזמנה למערכת הנהלת החשבונות לצורך הפקת תעודת משלוח, חשבונות וקבלה.	Web API

API מבוסס COM

דרישות מוקדמות להפעלה מוצלחת של ה-API

להלן רשימה של דרישות מוקדמות בכדי שיהיה ניתן להפעיל בהצלחה את ה-API של כספית:

- ❖ התקנה תקינה של כספית.
כלומר כספית נפתחת באופן תקין וניתן להוסיף לה לקוחות, חשבונות וכו'.
חשוב לבדוק גם הדפסות. כלומר שניתן להדפיס באופן תקין מכספית.
אחת המשמעויות היא שיש התקנה של Access 2003 או מאוחר יותר על המחשב.
- ❖ התקנה של CaspitAPI32.dll – זהו המימוש של ה-API עבור גרסה 3.2 של כספית. מאחר שהקומפוננטה היא ActiveX יש להריץ עליה regsvr32.
- ❖ התקנה של regobj.dll – קומפוננטה של מיקרוסופט לגישה ל-registry. יש להריץ עליה regsvr32.

❖ רישוי מתאים:

○ לצורך ההפעלה של כספית API COM - רישיון להפעלה של כספית שמכיל הרשאה לשימוש ב-API. אם תנסה להשתמש ב-API כשהרישיון של כספית לא מכיל הרשאה לשימוש ב-API חלק מהפעולות של ה-API יחסמו. לדוגמה: לא תוכל להציג מסמכים או להדפיס אותם.

○ לצורך ההפעלה של Web API – רישיון הפעלה לכספית שכולל הרשאה לשימוש ב-Caspit Web.

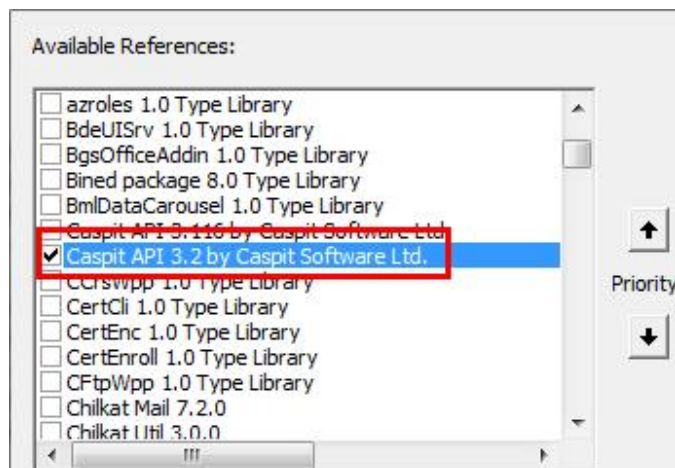
מה הטכנולוגיה של ה-API?

ה-API ממומש ב-VB6 ומקומפל בתור קומפוננטה ActiveX.

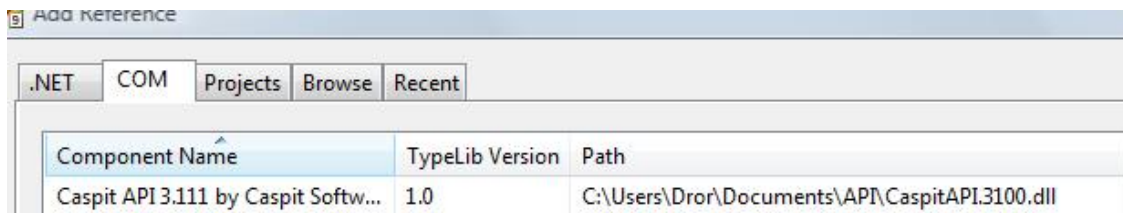
לאחר הרישום של הקומפוננטה על המחשב, מתחברים אליה כמו לכל אובייקט COM ומשתמשים במחלקות ובמתודות שהקומפוננטה חושפת.

כיצד מתחברים ל-API?

- שמה של הקומפוננטה הוא Caspit API 3.2 By Caspit Software Ltd.
- ב-VB ובתוכנות שיש בהן VBA (אקסס, אקסל וכו') מוסיפים את ה-REFERENCE הבא:



- ב-.Net מוסיפים Reference מוג COM:



- בתוכנות אחרות, יש לבצע קישור המתאים לתוכנה.
- ניתן לקבל פרויקט הדגמה של השימוש ב-API. הפרויקט ממומש ב-VB6.

הפעלה ב-XML

בכדי להקל עת השימוש ב-API פעולות רבות ממומשות בשתי דרכים:

- יצירה או שליפה של האובייקט באמצעות מתודה ו-CLASS
- יצירה או שליפה של האובייקט באמצעות XML.

לדוגמה, ה-API מאפשר לך ליצור חשבונית מס הן על-ידי `CreateInvoice(.. parameters ..)` והן על-ידי `CreateInvoiceXML(XML_string)`

במקביל ניתן לשלוף חשבונית מס על-ידי `GetInvoiceByNumber` שמחזיר אובייקט מסוג `Invoice` והן על-ידי `GetInvoicesByQueryXML(ilInvoiceNumber As Long)` שמחזיר XML ובו הפרטים של החשבונית.

הפעלה של ה-API ב DebugMode

האובייקט של ה-API מכיל את ה- `property` הבוליאני `DebugMode`. ברירת המחדל היא `False`. אם תפעיל את ה-API כאשר `DebugMode = True` תבצענה הפעולות הבאות:

- ה-API יצור את הקובץ `casplitapi3.212.32.txt` בתיקיה של כספית. שם הקובץ מורכב מ-`casplitapi` וממספר הגרסה של ה-API. ה-API יכתוב לקובץ פרטים רבים על ההפעלה שלו. לדוגמה:

```

05/06/2008 17:59:38 Caspit.GetInvoicesByQueryXML iInvoiceNumber 10000
-----
05/06/2008 18:02:08 Caspit.GetInvoicesByQueryXML iInvoiceNumber 10000
-----
05/06/2008 18:02:08 Caspit.GetInvoicesByQueryXML XML returned <Casplit>
<Document type="Invoice"><CustomersID>369</CustomersID><BusinessName>טסגרות שגול</BusinessName><Address1>ג"ל
8</Address1><Address2>4</Address2><City>ת"א</City><PostalCode>91919</PostalCode><ContactDetails>
בזנה שגול</ContactDetails><Phone1>03-626778</Phone1>
78</Phone1><Phone2></Phone2><Fax></Fax><TrxCodesID>13</TrxCodesID><DocDate>2008-06-
05T00:00:00</DocDate><DocComments></DocComments><DocDetails></DocDetails><Total1>128</Total1><RebatePercent>10<
/RebatePercent><RebateAmount>12.8</RebateAmount><RoundTotal>True</RoundTotal><RebateRoundAmount>.0485</RebateRo
undAmount><TotalBeforeVat>115.1515</TotalBeforeVat><ChargeVAT>True</ChargeVAT><VAT>15.5</VAT><VATAmount>17.8485
</VATAmount><GrandTotal>133</GrandTotal><StatusID>0</StatusID><ReceiptNumbers></ReceiptNumbers><LineItems><Line
Item number="1"><ItemDescription>מ עץ (מ'פס')
</ItemDescription><ItemQuantity>20</ItemQuantity><UnitPrice>2.9</UnitPrice><Currency>ש</Currency><CurrencyRate>
1</CurrencyRate><ExtendedPrice>58</ExtendedPrice></LineItem></LineItems></Document>
</Casplit>

```

משמאל לימין: תאריך ושעה, הפונקציה, הפרמטר, הערך של הפרמטר.

- כספית תפתח את הקובץ `casplitapi.txt` בתיקיה של כספית, ותרשום בו פרטים על ההפעלה שלה.

אם אתה מנסה להציג מסמכים בכספית, וכספית נפתחת ומיידי נסגרת, הפעל את ה-API במוד DEBUG וסביר שתראה בקובץ את הרשומה הבאה:

```

25/03/2011 20:00:34 modCasplitAPI.APIInit Started: iBusinessID = 1, g_blnIsAPIDebug = True
-----
25/03/2011 20:00:34 modLicence.CheckLegal Started
-----
25/03/2011 20:00:34 modLicence.CheckLegal Finished
-----
25/03/2011 20:00:34 modCasplitAPI.APIInit CasplitAPI=False
-----
25/03/2011 20:00:34 modCasplitAPI.APIInit API-רישיון ההפעלה אינו כולל הפעלה של ה

```

כל הזכויות שמורות © כספית תוכנה בע"מ

- כאשר ה-API מפעיל את כספית (לדוגמה להדפסה של מסמך) הוא יפתח את כספית במסך מלא. ברירת המחדל היא לפתוח את כספית במסך מוסתר.

משלוח מייל

ה-API מאפשר לשלוח מסמכים במייל מכספית.

בכדי לשלוח את המייל, ה-API משתמש ב-:


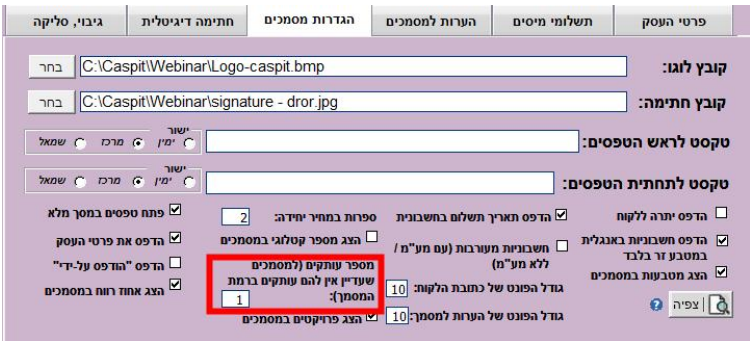
- מדפסת pdfFactory. [יש התקנה חנימית](#). אם אתה רוצה להסיר את ה-Evaluation Text שהמדפסת מדפיסה בתחתית המסמכים, רכוש רישיון למדפסת.
- אובייקט Dimac.jmail. לכן חייבים [להתקין גם את Dimac w3Jmail](#) (ההתקנה חנימית).

שליטה במספר העותקים המודפסים (3.2)

כאשר מדפיסים מסמכים דרך ה-API הוא מדפיס באופן הבא:

- אם המסמך עדיין לא הודפס אז יודפס מקור + X עותקים.
- אם המסמך כבר הודפס אז יודפס עותק אחד.

X נקבע באופן הבא:

שם המסמך	מספר העותקים כשמדפיסים מקור
חשבונית	לפי מה שהשתמש קבע בטופס חשבונית: 
חש/קבלה	לפי מה שהשתמש קבע בטופס חשבונית/קבלה
הזמנת עבודה	לפי מה שהשתמש קבע בטופס הזמנת עבודה
חשבונית עסקה	לפי מה שהשתמש קבע בטופס חש/עסקה
תעודת משלוח	לפי מה שהשתמש קבע בטופס תעודת משלוח
הצעת מחיר	לפי מה שהשתמש קבע בטופס הצעת מחיר
הזמנת רכש (כספית+מלאי)	לפי מה שהשתמש קבע בטופס הזמנת רכש
חשבון (כספית+עמותות)	לפי מה שהשתמש קבע בטופס חשבון
חשבון/קבלה (כספית+עמותות)	לפי מה שהשתמש קבע בטופס חשבון/קבלה
פיקדון (כספית+עוד)	לפי מה שהשתמש קבע בטופס פיקדון
יתר המסמכים	לפי מה שהשתמש קבע בפרטי העסק, לשונית הגדרת מסמכים: 

הבדלים בין ה-API של כספית 3.1 ל-API גרסה 3.2

ההבדל המשמעותי בין גרסה 3.1 של כספית לגרסה 3.2 של כספית הוא האפשרות להפעיל מספר עסקים בכספית אחת. מהזווית של הפעלת ה-API יש דרישה לציין לאיזה מספר עסק בכספית מתייחסת הפעולה (יצירת חשבונית, עדכון לקוח וכו').

בכדי להקל על ההתאמה לקוד קיים הוספו ל-API בשם BusinessID. לאחר יצירת האובייקט CaspitAPI אפשר לתת ערך ל-property בכדי לציין לאיזה מספר עסק מתכוונים.

ערך ברירת מחדל של ה-property הוא 1. לכן קוד קיים ימשיך לעבוד מול עסק מספר 1 ללא שינוי.

ה-ID של רשומות בכספית 3.2

ה-ID של רשומות בכספית 3.2 אינו קבוע ויכול להשתנות! לכן, רצוי מאד להסתמך על

1. מספר הרשומה: מספר חשבונית, מספר לקוח, ולא על ה-ID שלה.

2. המפתח הזר שנותנים ללקוחות.

שם לב: ה-ID ישתנה בכל פעם שמייבאים נתונים לכספית.

הסיבה: הנתונים של כל העסקים נשמרים בטבלאות משותפות (יש טבלה אחת לחשבוניות, טבלה אחת ל-חש/קבלה וכו'). יש צורך למפות את ה-ID של רשומות של עסקים שונים, כשמייבאים אותם לכספית, בכדי למנוע "התנגשויות" של ערכים זהים.

ה-ID של רשומות שעברו מיפוי יהיה מספר שלילי ארוך:

iCustomersID	datissuDate	datxDate	datCreation	iBusin	invoice	invoicess
Mr. Li Huyi	23/03/2011	22/03/2011	23/03/2011 00:09:00	2	10001	26
יהונתן מאיו	18/03/2011	18/03/2011	18/03/2011 18:59:59	6	10001	3
Mr. Li Huyi	22/03/2011	22/03/2011	22/03/2011 09:59:25	2	10000	25
יהונתן מאיו	18/03/2011	18/03/2011	18/03/2011 16:57:42	6	10000	2
אולפני הקלטה טופו	14/03/2011	14/03/2011	14/03/2011 10:03:34	1	390	-10000029
דפנה לבנון	14/03/2011	14/03/2011	14/03/2011 09:22:32	1	389	-10000028
הישיבה הגבוהה אור עציון	13/03/2011	13/03/2011	13/03/2011 13:01:21	1	388	-10000026
אמ"ת רחובות בנים כלי	10/03/2011	10/03/2011	10/03/2011 11:53:13	1	387	-10000025
מרס נריה	10/03/2011	10/03/2011	10/03/2011 11:13:49	1	386	-10000024
הישיבה הגבוהה אור עציון	09/03/2011	09/03/2011	09/03/2011 09:25:50	1	385	-10000022
המרכז לדיי וטונרים	08/03/2011	08/03/2011	08/03/2011 14:10:00	1	384	-10000021
יוסף סעידוב	08/03/2011	08/03/2011	08/03/2011 13:56:14	1	383	-10000020
רועי אשרי	08/03/2011	08/03/2011	08/03/2011 11:01:15	1	382	-10000019

שינויים נוספים

הערות	השינוי ב-API	טבלה בכספית
בגרסה 3.1 מספר הלקוח היה פשוט ה-ID שלו בטבלת לקוחות. בגרסה 3.2 נאלצנו להפריד בין ה-ID של הלקוח לבין מספר הלקוח	לאובייקט Customer נוסף השדה CustomerNum בנוסף הוספנו את השדה כפרמטר במתודה של יצירת לקוח. אם הערך שלו הוא 0 אז ה-API יתן ללקוח את מספר הלקוח אוטומטית.	לקוחות – נוסף השדה מספר לקוח
	לאובייקט Customer נוספו השדות Extra1 (הערות2) ו-Extra2 (הערות3) בנוסף הוספנו את השדות כפרמטרים במתודה של יצירת לקוח.	לקוחות – נוסף השדות: הערות2, הערות3
בגרסה 3.1 מספר המחירון היה ה-ID שלו בטבלת מחירונים.	השדה iListNum מכיל את ה-ID של המחירון ולא את המספר שלו.	לקוחות – מספר המחירון

כל הזכויות שמורות © כספית תוכנה בע"מ

בגרסה 3.2 נאלצנו להוסיף לטבלת מחירונים את iListsID ולהפריד בין ה-ID של המחירון מהמספר שלו (= מבחינת המשתמש המחירונים ממוספרים 1, 2, 3, ..)		
השינוי נדרש בעקבות ההתאמה של כספית ל-PCN874.	נוסף השדה vcOsekMorshe שמכיל את מספר העוסק המורשה של הלקוח כפי שהוא כתוב במסמך	חשבונית, חש/קבלה – מספר העוסק מורשה של הלקוח
אחד הפרטים החשובים של חשבונית, חש/קבלה ומסמכים אחרים הוא סיווג התנועה שלו. בכספית 3.1 ה-ID של סיווג התנועה 3-הכנסות/מכירות היה קבוע. בכספית 3.2 ה-ID משתנה מעסק לעסק.	צריך לשלוף את ה-ID של סיווג התנועה מכספית, כי הוא יכול להשתנות מעסק לעסק. ה-ID גם משתנה בזמן היבוא של נתונים מכספית 3.1 לכספית 3.2!	סיווגי תנועות
	נוסף האובייקט PriceList ונוספה מתודה לשליפה של כל המחירונים	מחירונים – iListNumber הופרד מ-iListsID
השדה מאפשר לקבוע את ה-ID של הפריט. ה-ID חשוב בכספית+עוד ובכספית+מלאי	נוסף השדה iltemsID ברירת מחדל שלו היא 1, כך שקוד קיים ימשיך לעבוד.	שורות חשבונית, חש/קבלה וכו'
הבחירה בפריט חשובה בכספית+מלאי לצורך עדכון המלאי, ובכספית+עוד לצורך חישוב נכון של הוצאות הלקוח.	נוספה היכולת לשלוף מערך של פריטים	פריטים
ראה הסבר לעיל.	ה-API משתמש במה שהמשתמש קבע בכספית (ומתעלם מה-REGISTRY)	הדפסות – מספר ההעתקים המודפסים
	נוספה יכולת להדפיס חשבונית באנגלית וקבלה באנגלית	הדפסות

הערה לגבי עדכון לקוח לפי מספר

סימון של חידושים ב-API בגרסה 3.2

סימנו חידושים בגרסה 3.2 כך: (3.2).

מפת כיסוי של ה-API

להלן טבלה המציגה את המסמכים/אובייקטים העיקריים בכספית ויכולת היצירה והשליפה שלהם דרך ה-API:

שם המסמך/יכולת	יצירה עם פרמטרים	יצירה עם XML	שליפה של אובייקט יחיד	שליפת מערך של אובייקטים	שליפת נתונים ב-XML	הצגה בכספית	הדפסה עברית בלבד אלא אם צוין אחרת
חשבונית עסקה	כן	כן				כן	כן
חשבונית מס	כן	כן		כן	כן – לפי מספר חש' חש'	כן	כן (גם אנגלית (3.2))
חש/קבלה	כן	כן		כן	כן	כן	כן
קבלה	כן	כן				כן	כן (גם אנגלית (3.2))
חש' זיכוי	כן	כן				כן	כן
תעודת החזר	כן	כן				כן	כן
לקוח	כן + עדכון			כן	כן		לא
שליפת לקוח לפי מפתח זר	---	---		כן	כן		לא
שליפת יתרת לקוח לפי מפתח זר וגם לפי ID של הלקוח (3.2)			כן – מחזיר DOUBLE	---	לא		
שליפה של יתרת לקוחות עם/ללא חשבוניות עסקה (3.2)	---	---	---	---	כן	---	---
סווג תנועה	---	---	לא	כן		---	---
שליפה של אחוז המע"מ	---	---	כן	---	---	---	---
שליפה של שערי החליפין של מט"ח	---	---	כן	---	---	---	לא
מחירוים (3.2)	לא	לא	לא	כן	לא	---	לא
פריטים (כספית+מלאי, כספית+עוד) (3.2)	לא	לא	לא	כן	לא	---	לא

כל הזכויות שמורות © כספית תוכנה בע"מ

הדפסה עברית בלבד אלא אם צוין אחרת	הצגה בכספית	שליפת נתונים ב-XML	שליפת מערך של אובייקטים	שליפה של אובייקט יחיד	יצירה עם XML	יצירה בכספית עם פרמטרים	שם המסמך/יכולת
---	כן – במובן של הפעלה של כספית	---	---		---	---	ביצוע של גיבוי של כספית (3.2)
---	כן – במובן של הפעלה של כספית	---	---		---	---	משלוח מסמכים במייל
---	---	לא	לא	כן	לא	---	שליפה של ערך של Registry כספית (3.2)
---	---	לא	לא	כן	לא	---	שליפה של ערך מטבלת הפרמטרים של כספית (3.2)

מקרא:

--- - לא רלוונטי

(3.2) – חדש או עודכן בגרסה 3.2

תכונות Properties

המחלקה הראשית של ה-API היא Caspit. לאובייקט התכונות הבאות:

- ❖ `objCaspit.CaspitPath` – שם הקובץ (כולל מסלול) של כספית. לדוגמה: `c:\program files\caspit\caspit30.mde`
ה-API מאתחל את עצמו למסלול של כספית כפי שרשם ב-Registry תחת `.HKLM\Software\Caspit`
- ❖ `objCaspit.AccessPath` – שם הקובץ (כולל מסלול) לאקסס. לדוגמה: `C:\Program Files\Microsoft Office\OFFICE11\msaccess.exe`
ה-API מאתחל את עצמו למסלול ל Access 2003 לפי מה שרשום ב-registry.
- ❖ `DebugMode` – בוליאני. ברירת מחדל: `False`. אם הערך הוא `True` ה-API יפעל במודד באג. ראה הסבר לעיל.
- ❖ `MailServerPassword`, `MailServerUserName`, `MailServer.SendDocByMail` ו `SendMail` – משמש את ה-API עבור המתודות
- ❖ `PDFTimeout` – הזמן (במילישניות) להמתין למדפסת PDF כאשר משתמשים במתודה `SendDocByMail`. לעיתים יש צורך לאפשר יותר זמן למדפסת לסיים את ההפקה של הקובץ.
- ❖ `Version` - מחזיר את מספר הגרסה של ה-API.
- ❖ `BusinessID` – מספר העסק בכספית. ברירת המחדל היא 1. (חדש ב-3.2)
- ❖ `BackupTimeOutSeconds` – הזמן (בשניות) להמתין לכספית כאשר היא מבצעת Backup. (חדש ב-3.2)

לקוחות

יצירה של לקוח – CreateCustomer

יצירה של לקוח חדש בכספית.

```
Function CreateCustomer(strBusinessName As String,  
    strOsekMorshe As String_ ,  
    strContactFirstName As String, strContactLastName As String ,  
    strAddress1 As String, strAddress2 As String, strCity As String,  
    strPostalCode As String_ ,  
    strPhone1 As String, strPhone2 As String, strFax As String ,  
    strEMail As String ,  
    strComments As String,  
    curOpenBalance As Currency ,  
    strBankAccountNumber As String,  
    strBankBranch As String,  
    iBankID As Long,  
    iListNum As Long,  
    curCharge As Currency,  
    strCurrencyName As String ,  
    strForeignKey As String ,  
    Optional iPaymentTermsID As PaymentTerms,  
    Optional iPaymentDays As Long) As Long
```

שמות הפרמטרים מתארים אותם.

- ❖ iListNum – ID של המחירון של הלקוח (לכל לקוח אפשר לשייך מחירון אחד לכל היותר).
 - ❖ curCharge – העלות של שעת עבודה. פרמטר של כספית+יועצים שמיועדת למי שמחייב את הלקוחות שלו לפי שעות עבודה.
 - ❖ strCurrencyName – שם המטבע (EUR ,GBP ,USD ,NIS) שבו מחייבים את curCharge .
 - ❖ strForeignKey – המפתח של הלקוח במערכת החיצונית.
 - ❖ iPaymentTermsID – תנאי התשלום של הלקוח (מזומן, שטוף)
 - ❖ iPaymentDays – אם תנאי התשלום הם שטוף, מספר הימים לשטוף (30, 60 וכו').
- המתודה מחזירה את ה-ID Customer שנוצר בכספית.

יצירה של לקוח לפי XML – CreateCustomerXML

```
Function CreateCustomer(strXML As String) As Long
```

ראה פירוט של ה-XML בפרק המתאים.

המתודה מחזירה את ה-ID Customer שנוצר בכספית.

GetCustomerByForeignKey – שליפה של לקוח לפי מפתח זר

```
Function GetCustomerByForeignKey(strForeignKey As String) As Customer
```

❖ strForeignKey – המפתח הזר של הלקוח.

המתודה מחזירה אובייקט מסוג Customer. ה-Customer הוא אובייקט ששייך ל-API.

GetCustomerByForeignKeyXML – שליפה של לקוח לפי מפתח זר

```
Function GetCustomerByForeignKeyXML(strForeignKey As String) As String
```

❖ strForeignKey – המפתח הזר של הלקוח.

הפונקציה מחזירה את פרטי הלקוח ב-XML. ראה פירוט של מבנה ה-XML בפרק המתאים.

GetCustomerBalanceByID – שליפה של יתרת הלקוח לפי המפתח בכספית

```
Function GetCustomerBalanceByID(iCustomerID As Long) As Double
```

❖ iCustomerID – המפתח של הלקוח בכספית.

הפונקציה מחזירה את יתרת הלקוח. היתרה של הלקוח היא סכום של כל החשבונות, קבלות, תעודות החזר, חשבונות זיכוי וכו'.

סכום חיובי מייצג יתרת זכות (העסק חייב כסף ללקוח), סכום שלילי מייצג יתרת חובה (הלקוח חייב כסף לעסק).

חשבונות עסקה מחושבת לפי הסכום האפקטיבי שלה. הסכום האפקטיבי של חשבונות זיכוי הוא סכום החשבונות פחות הסכום של כל חשבונות המס שהופקו ממנה.

GetCustomerBalanceByForeignKey – שליפה של יתרת הלקוח לפי מפתח זר

```
Function GetCustomerBalanceByForeignKey(strForeignKey As String) As Double
```

❖ strForeignKey – המפתח הזר של הלקוח.

הפונקציה מחזירה את יתרת הלקוח. היתרה של הלקוח היא סכום של כל החשבונות, קבלות, תעודות החזר, חשבונות זיכוי וכו'.

סכום חיובי מייצג יתרת זכות (העסק חייב כסף ללקוח), סכום שלילי מייצג יתרת חובה (הלקוח חייב כסף לעסק).

חשבונות עסקה מחושבת לפי הסכום האפקטיבי שלה. הסכום האפקטיבי של חשבונות זיכוי הוא סכום החשבונות פחות הסכום של כל חשבונות המס שהופקו ממנה.

GetCustomers – שליפה של כל הלקוחות של כספית

```
V = objCaspit.GetCustomers()
```

המתודה מחזירה מערך של CaspitAPI.Customer. ה-Customer הוא אובייקט ששייך ל-API.

GetCustomersXML – שליפה של כל הלקוחות של כספית

המתודה מחזירה את פרטי הלקוחות ב-XML. ראה פירוט של מבנה ה-XML בפרק המתאים.

Customer – אובייקט לקוח

האובייקט מכיל את התכונות הבאות:

- ❖ CustomersID – ה-ID של הלקוח בכספית
- ❖ BusinessName – שם העסק
- ❖ פרטי חשבון הבנק של הלקוח:
- BankAccountNumber – מספר החשבון
- BankBranchNumber, יכול להיות גם שם הסניף
- BankID – מספר הבנק לפי הרשימה שבנספח א'
- ❖ CreditCard, ExpDate – מספר כרטיס אשראי של הלקוח ותאריך התוקף. **בוטל בגרסה 3**
- ❖ ForeignKey – המפתח הזר של הלקוח במערכת שיצרה אותו. זה המפתח שנשלח ב- CreateCustomer. **ללקוחות שנוצרו בתוך כספית אין מפתח זר.**
- ❖ PaymentTerms ו-PaymentDays – תנאי התשלום של הלקוח. ישנם שני סוגים של תנאי תשלום: מזומן ושוטף+X. אם תנאי התשלום הם שוטף אז יש צורך לציין את מספר הימים ב-PaymentDays.
- ❖ PriceListNum – מספר המחירון של הלקוח (כספית+מלאי).
- ❖ Charge ו-ChargeCurrency – העלות של שעת עבודה (כספית+יועצים)

וכל:

- Members of 'Customer'
- Address1
 - Address2
 - BankAccountNumber
 - BankBranchNumber
 - BankID
 - BusinessName
 - Charge
 - ChargeCurrency
 - City
 - ContactFirstName
 - ContactLastName
 - CustomersID
 - EMail
 - Fax
 - ForeignKey
 - IsChanged
 - OsekMorshe
 - PaymentDays
 - PaymentTerms
 - Phone1
 - Phone2
 - PostalCode
 - PriceListNum

UpdateCustomerByForeignKey -- עדכון של לקוח לפי מפתח זר

```
Function UpdateCustomerByForeignKey(strBusinessName As String,
strOsekMorshe As String, _
strContactFirstName As String, strContactLastName As String, _
strForeignKey As String, _
```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

Optional strAddress1 As String, Optional strAddress2 As String,
Optional strCity As String, Optional strPostalCode As String, _
Optional strPhone1 As String, Optional strPhone2 As String, Optional
strFax As String, _
Optional strEMail As String, _
Optional strBankAccountNumber As String, Optional strBankBranch As
String, Optional iBankID As Long, _
Optional iListNum As Long, Optional curCharge As Currency, Optional
strCurrencyName As String _
)

```

❖ strForeignKey - המפתח הזר של הלקוח.

הערה: לא ניתן לעדכן את strForeignKey

המתודה מחזירה את ה- Customer ID של הלקוח בכספית.

UpdateCustomerByForeignKeyXML -- עדכון של לקוח לפי מפתח זר

```
Function UpdateCustomerByForeignKeyXML(strXML As String) As Long
```

❖ strForeignKey - נכלל ב-XML. המפתח הזר של הלקוח. לא ניתן לעדכן את strForeignKey

❖ ראה פירוט של מבנה ה-XML בפרק המתאים.

המתודה מחזירה את ה- Customer ID של הלקוח בכספית.

GetCustomersBalance – שליפה של יתרת לקוחות

```
Function GetCustomersBalance(datDateTo As Date, Optional
blnIncludingInvoicesIska As Boolean = False, Optional lngTimeOutSeconds
As Long = 20) As String
```

datDateTo – כספית מאפשרת לחשב את יתרת הלקוחות ליום מסוים. המימוש על-ידי כך שכוללים בחישוב מסמכים שהתאריך שלהם קטן-שווה לתאריך המבוקש.

blnIncludingInvoicesIska – האם לכלול בחישוב חשבוניות עסקה.

lngTimeOutSeconds – החישוב הוא מסובך ויש חשש "לתקיעה" בעקבות נפילה של כספית. אם החישוב לא יסתיים עד lngTimeOutSeconds שניות, ה-API יעלה שגיאה.

המתודה מחזירה XML ובו פירוט של יתרת הלקוחות ליום המבוקש. ה-XML מכיל

- את הערך ToDate שהוא הערך עבורו חושבה יתרת הלקוחות.

- אינדיקציה האם היתרה כוללת חשבוניות עסקה.

דוגמה ל-XML שמחזירה המתודה

```

<Casplit>
<ToDate>26/08/2011</ToDate> ← התאריך שעבורו חושבו היתרות
<IncludesIska>True</IncludesIska> ← האם חישוב היתרה כולל חשבוניות עסקה
<Customers>
<Customer><CustomersID>-
110000004</CustomersID><CustomerNumber>3</CustomerNumber><BusinessName>
האלון בר גיל</BusinessName><Balance>-11544</Balance></Customer>

<Customer><CustomersID>27</CustomersID><CustomerNumber>10</CustomerNumber>
<BusinessName>האלון בר גיל</BusinessName><Balance>-1160</Balance></Customer> ←
סכום שלילי מצויין חוב של הלקוח

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

<Customer><CustomersID>4</CustomersID><CustomerNumber>4</CustomerNumber>
<BusinessName>ער-י בת-יה</BusinessName><Balance>-
1013648.4</Balance></Customer>

<Customer><CustomersID>6</CustomersID><CustomerNumber>6</CustomerNumber>
<BusinessName>ער-י לחמני לודו</BusinessName><Balance>-
8776.8</Balance></Customer>

<Customer><CustomersID>5</CustomersID><CustomerNumber>5</CustomerNumber>
<BusinessName>ער-י תות</BusinessName><Balance>-
2886.4</Balance></Customer>

</Customers>
</Casplit>

```

דוגמת קוד לשליפת יתרת לקוחות

הקוד הבא שולף את יתרת הלקוחות כולל חשבוניות/עסקה ליום 25/2/2012. ה-TIMEOUT יקרה אחרי 40 שניות.

```

Dim S As String
Dim objCasplit As CasplitAPI.Casplit

Set objCasplit = New CasplitAPI.Casplit
S = objCasplit.GetCustomersBalance("25/2/2012", True, 40)
Set objCasplit = Nothing

```

RefreshCustomerBalance – עדכון של יתרת הלקוחות (3.2)

המתודה מעדכנת את הטבלאות tblCustomerBalance ו-tblCustomerBalancelaska שבה שמורים הערכים של יתרת הלקוחות. בכדי להאיץ את הצגת המסך של יתרת לקוחות בכספית, הנתונים שמורים בטבלאות.

ה-API בודק את התאריך שעבורו חושבו הנתונים ואם יש צורך, משתמש במתודה RefreshCustomerBalance בכדי לעדכן את הנתונים. המשמעות: אין צורך לקרוא ל-RefreshCustomerBalance באופן ישיר.

תהליך העדכון הוא תהליך מסובך ומלווה בחישובים רבים. בכדי שתוכל לדעת מתי התהליך מסתיים כספית מעדכנת את ה-registry בסיום העדכון. הנה דוגמת קוד כיצד אפשר להמתין לסיום הפעולה.

```

Dim objCasplit As CasplitAPI.Casplit
Dim V As String

Set objCasplit = New CasplitAPI.Casplit
V = objCasplit.GetCasplitRegSetting("CustomerBalanceLastDateComputed", Now)

objCasplit.RefreshCustomerBalance "31/1/2011"

'Wait 1.2 seconds and check of finished
While objCasplit.GetCasplitRegSetting("CustomerBalanceLastDateComputed", Now) = V
    Sleep 1200
    DoEvents
Wend

```

שליפה של יתרת הלקוחות ישירות מכספית

הערכים של יתרת הלקוחות של כל העסקים שמורים בשתי טבלאות: tblCustomerBalance ו-tblCustomerBalancelaska. אתה יכול לשלוף את הערכים ישירות מהטבלאות.

שם לב, שהערכים נכונים ליום מסוים. אתה יכול לשלוף את היום עבורו חושבו הערכים כך:

```
V = objCaspit.GetSystemParameter("qryCustomerBalanceDate")
```

אובייקטים משניים

סיווג תנועה - TrxCode

בכדי ליצור חשבונית בכספית צריך לסווג אותה. הסיווג הסטנדרטי הוא "מכירות" אבל המשתמש יכול להגדיר סיווגים כרצונו. בכדי לאפשר ל-API לסווג את החשבונית הוספנו את המחלקה TrxCode:

- PLCodesID
- TrxCode
- TrxCodeDescription
- TrxCodesID

- ❖ PLCodesID – סיווג רווח והפסד בכספית. די לדעת ש-PLCodesID של כל סיווגי המכירות יהיה 1. זה דרוש בכדי להציג למשתמש רשימה מתאימה של סיווגי תנועה לבחור מהם במסך יצירת חשבונית.
- ❖ TrxCode – המספר שהמשתמש בחר לסיווג התנועה (לדוגמה: 102).
- ❖ TrxCodeDescription – תיאור הסיווג. לדוגמה: הכנסות מהרצאות.
- ❖ TrxCodesID – ה-ID של הסיווג בכספית.

שליפה של כל סיווגי התנועות – GetTrxCodes

```
V = objCaspit.GetTrxCodes()
```

המתודה מחזירה מערך של CaspitAPI.TrxCode.

שליפה של כל סיווגי התנועות – GetTrxCodesXML (3.2)

```
S = objCaspit.GetTrxCodesXML()
```

המתודה מחזירה מחרוזת ובה XML שכולל את כל סיווגי התנועות שבכספית.

לדוגמה, הנה סיווג תנועה בכספית:

204	קניות ללא מע"מ	תצרכת חומרים	לא משתתף בדו"ח מע"מ	אין	מלא
-----	----------------	--------------	---------------------	-----	-----

המתודה תחזיר את ה-XML הבא (ה-".."). מסמנים שיסגרו ב-XML סיווגי תנועות נוספים):

```
<Caspit>
  <TrxCodes>
    ...
    <TrxCode>
      <iTrxCodesID>75</iTrxCodesID>
      <TrxCodeNumber>204</TrxCodeNumber>
      <TrxCodeDescription>קניות ללא מע"מ</TrxCodeDescription>
      <PLCodesID>2</PLCodesID>
    </TrxCode>
    ...
  </TrxCodes>
</Caspit>
```

תנאי תשלום PaymentTerms

תנאי התשלום של הלקוח. ישנם שני סוגים של תנאי תשלום: מזומן ושוטף+X. אם תנאי התשלום הם שוטף אז יש צורך לציין את מספר הימים ב-PaymentDays.

Public Enum PaymentTerms

Net = 1

Cash = 2

End Enum

חשבוניות

CaspitAPI.Invoice אוביקט חשבונית

האובייקט מייצג חשבונית מס. יש לו התכונות הבאות:

Members of 'Invoice'	
blnChargeVAT	
blnRebateRound	
curGrandTotal	
curRebateAmount	
curRebateRoundAmount	
curTotal1	
curTotalBeforeVat	
curTotalNoVat	
curVAT	
datDueDate	
datxDate	
fltRebatePrct	
iCasesID	
iCustomersID	
iInvoiceNumber	
iStatusID	
iTxCodesID	
strReceiptNumbers	
varInvoiceLines	
vcAddress1	
vcAddress2	
vcBusinessName	
vcCity	
vcComments	
vcContactDetails	
vcDetails	
vcFax1	
vcPhone1	
vcPhone2	
vcPostalCode	

blnChargeVAT	האם לגבות מע"מ בחשבונית
blnRebateRound	האם לעגל את המחיר הסופי
curGrandTotal	המחיר כולל מע"מ
curRebateAmount	ההנחה המסחרית שניתנה ללקוח בש"ח. אם אין הנחה: 0
curRebateRoundAmount	הנחת העיגול בש"ח (אם אין הנחת עיגול: 0)
curTotal1	סה"כ שורות החשבונית (לפני הנחה מסחרית ולפני הנחת עיגול)
curTotalBeforeVAT	סה"כ של החשבונית לפני מע"מ (אחרי הנחה מסחרית ואחרי הנחת עיגול)








כל הזכויות שמורות © כספית תוכנה בע"מ

curTotalNoVAT	סה"כ שורות החשבונית שאין עליהן מע"מ – רלוונטי לחשבוניות מעורבות שיש בהן שורות חשבונית שיש עליהן מע"מ ושורות חשבונית שאין עליהן מע"מ
curVAT	סכום המע"מ
datDueDate	מועד התשלום
fltRebatePrct	ההנחה המסחרית באחוזים
iCasesID	ה-ID של הפרויקט (כספית+)/(תיק (בכספית+עוד) שהחשבונית משייכת אליו.
iCustomersID	ה-ID של הלקוח של החשבונית
iInvoiceNumber	מספר החשבונית
iStatusNumber	הסטטוס של החשבונית: 0 – חדשה שלא הודפסה, 2 – הודפסה ונעולה לשינויים, 3 – מבוטלת.
iTrxCodesID	ה-ID של סיווג התנועה של החשבונית
strReceiptNumbers	מספרי הקבלות שמשייכות לחשבונית. מופרדים בפסיק. לדוגמה: "30005,30010,30003".
varLinItems	מערך של שורות החשבונית.
	פרטים של הלקוח לקוחים מהחשבונית. יתכן שפרטי הלקוח בטבלת לקוחות שונים כי עדכנו את הרשומה של הלקוח אחרי הפקת החשבונית.
vcOsekMorshe	מספר העוסק מורשה של הלקוח כפי שרשום בחשבונית. (3.2)

CaspitAPI.LinItem – שורת חשבונית

האובייקט מייצג שורת חשבונית. יש לו התכונות הבאות:

Members of 'LinItem'

-  ChargeVAT
-  CurrencyRate
-  ExtendedPrice
-  ItemCurrency
-  ItemDescription
-  ItemQuantity
-  UnitPrice

- ❖ ChargeVat – בוליאני. האם לגבות מע"מ על שורת החשבונית.
- ❖ ExtendedPrice – "מחיר" השורה. כלומר: $ItemQuantity * UnitPrice$.
- ❖ ItemCurrency – המטבע שבו נמכר הפריט. ערכים אפשריים: NIS, USD, GBP, EUR
- ❖ CurrencyRate – שער המטבע. לדוגמה. עבור דולר השער יהיה 3.45
- ❖ ItemDescription – תיאור הפריט.
- ❖ UnitPrice – מחיר יחידה. אם אתה מחלץ את מחיר היחידה מהמחיר כולל מע"מ, רצוי מאד לשלוח את מחיר היחידה עם 4 ספרות אחרי הנקודה לכל הפחות. כספית תציג את מחיר היחידה מעוגל ל-2 ספרות (זו ברירת המחדל, המשתמש יכול לקבוע את מספר הספרות בעצמו), אבל תשתמש בכל הספרות

כל הזכויות שמורות © כספית תוכנה בע"מ

לחישוב המע"מ. זה ימנע תקלה אופיינית שבה המחיר כולל מע"מ שונה באגורה מהמחיר המקורי שממנו יצאת.

יצירה של חשבונית – CreateInvoice

```
Function CreateInvoice(iCustomersID As Long, datInvoiceDate As Date, strComments As String, fltRebatePercent As Double, curRebateAmount As Currency, blnRoundTotal As Boolean, fltVAT As Double, blnChargeVAT As Boolean, strInvoiceDetails As String, iTrxCodesID As Long, iInvoiceType As InvoiceTypes, varLineItems, [datDueDate As Date]) As Long
```

הפרמטרים הם:

- ❖ iCustomersID – ה-ID של הלקוח עבורו יוצרים את החשבונית
- ❖ strComments – הערות לחשבונית (מוצגות ללקוח). לדוגמה: הסחורה בבעלותנו עד לתשלום מלא של החשבונית.
- ❖ fltRebatePercent – הנחה באחוזים. לדוגמה: 9 פירושו 9%.
- ❖ curRebateNIS – הנחה בש"ח. כספית מאפשרת למשתמש להכניס הנחה באחוזים או בש"ח. ההנחה בש"ח גוברת על ההנחה באחוזים.
- ❖ blnRoundTotal – האם לעגל את הסכום לש"ח שלמים.
- ❖ blnChargeVAT – האם לגבות מע"מ. שם לב שבמקרה שלא גובים מע"מ סיווג התנועה יהפוך באופן אוטומטי ל מכירות ללא מע"מ ומתעלמים מסיווג התנועה ששלחת. זה דרוש בכדי שדו"ח מע"מ יצא תקין.
- ❖ strDetails – פרטים של החשבונית (אינם מוצגים ללקוח). לדוגמה: תשלום שני עבור פרויקט חיפה.
- iTrxCodesID – ה-ID של סיווג התנועה שבו רוצים לסווג את החשבונית. בכספית ישנם שני סיווגים בסיסיים לחשבוניות:

3	הכנסות/מכירות	הכנסות/מכירות	עסקאות ח"בות במע"מ	מלא	מלא
4	הכנסות/מכירות ממע"	הכנסות/מכירות	עסקאות פטורות ממע"מ	אין	מלא

○ אם אתה יוצר חשבונית עם מע"מ השתמש בסיווג מספר 3 (מצא את ה-ID בעזרת המתודה GetTrxCodes)

○ אם אתה יוצר חשבונית ללא מע"מ השתמש בסיווג מספר 4 (מצא את ה-ID בעזרת המתודה GetTrxCodes)

❖ varLineItems – מערך של Lineltem.

❖ datDueDate – התאריך לתשלום

הפונקציה מחזירה את מספר החשבונית שנוצרה.

יצירה של חשבונית על-ידי XML – CreateInvoiceXML

```
Function CreateInvoiceXML(strXML As String) As Long
```

ראה פירוט של מבנה ה-XML בפרק המתאים.

הפונקציה מחזירה את מספר החשבונית שנוצרה.

שליפה של חשבונית לפי מספר החשבונית

```
Function GetInvoiceByNumber(iInvoiceNumber As Long) As Invoice
```

ilInvoiceNumber – מספר החשבונית. לדוגמה: 10003

המתודה מחזירה אובייקט מסוג CaspitAPI.Invoice. לדוגמה:

```
Dim objCaspit As CaspitAPI.Caspit
```

```
Dim clsInvoice As CaspitAPI.Invoice
```

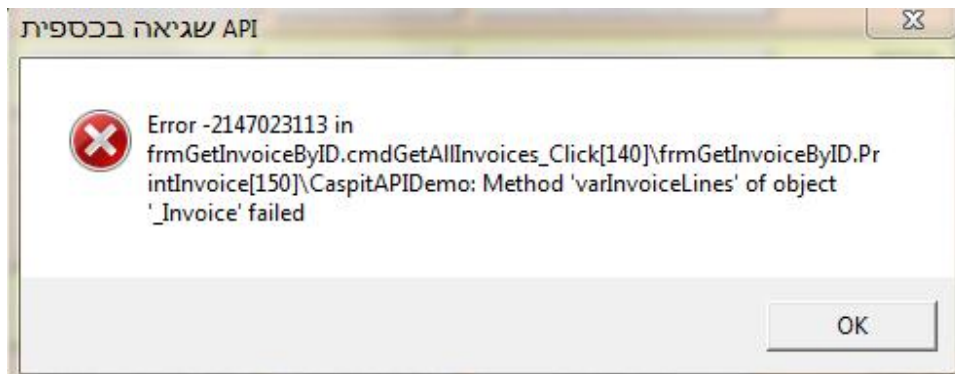
כל הזכויות שמורות © כספית תוכנה בע"מ

Initialize objCaspit ...

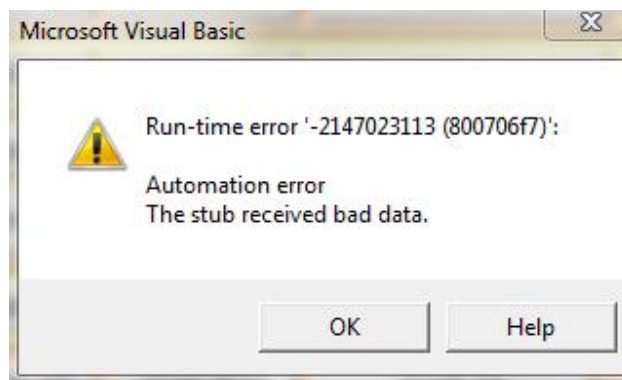
Set clsInvoice = objCaspit.GetInvoiceByNumber(10009)

שגיאה "0x800706F7 (The stub received bad data)"

ניגשים ל- clsInvoice.varInvoiceLines ומקבלים שגיאה דומה לזו:



או זו:



השגיאה מופיעה כאשר לחשבונית אין אף שורת חשבונית ולכן המערך של שורות חשבונית הוא ריק. נא ראה פיתרון אפשרי כאן: <http://support.microsoft.com/kb/975415>. לצערי, בבדיקות שלי (חלונות 7, 32 ביט) ההתקנה של ה-hot fix לא פתרה את הבעיה. יש דוגמה לקוד אפשרי בכדי לעקוף את הבעיה ב-API Demo frmGetInvoiceByNumber במתודה PrintInvoice.

שליפה של כל החשבוניות שיש בכספית – GetInvoices

```
Function GetInvoices() As Invoice()
```

המתודה מחזירה מערך של אובייקטים מסוג CaspitAPI.Invoice

DisplayInvoice --הצגה של חשבונית

המתודה מפעילה את כספית ומציגה את מסך החשבונית.

```
Sub DisplayInvoice(iInvoiceNumber as Long)
```

❖ מספר החשבונית (לדוגמה 10028) -- InvoiceNumber

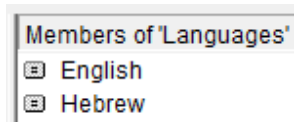
כל הזכויות שמורות © כספית תוכנה בע"מ

הדפסה של חשבונית – PrintInvoice

המתודה מדפיסה חשבונית בעברית או באנגלית.

```
Sub PrintInvoice(iInvoiceNumber As Long, Optional Lang As Languages = Hebrew)
```

- ❖ מספר החשבונית (לדוגמה 10028) -- iInvoiceNumber
- ❖ Lang – מספר השפה. רצוי להשתמש בקבועים שמוגדרים באנומריצה Languages.



ראה הסבר בקדמה על מספר העותקים שיודפסו עם המקור.

הערה חשובה: כאשר מדפיסים חשבונית מתוך מסך חשבונית בכספית, מודפס מקור ומייד אח"כ החשבונית ננעלת ומכאן והלאה אפשר להדפיס ממנה רק העתקים.

הסבר על הדפסה באנגלית

כספית מאפשרת להדפיס מסמכים באנגלית במט"ח מתורגם לש"ח ובמט"ח ללא תירגום.

המשתמש קובע האם להדפיס את המסמכים באנגלית במט"ח או בש"ח במסך פרטי העסק, לשונית הגדרת מסמכים, בשדה הדפס חשבוניות באנגלית במטבע זר בלבד.

שם לב, שאם המשתמש יצר חשבונית עם מטבעות מעורבים (פריטים בש"ח עם פריטים ביורו), אז כספית תדפיס את החשבונית באנגלית מתורגמת לש"ח, גם אם המשתמש קבע להדפיס במט"ח.

דוגמאות:

שם הפריט	מחיר פריט כולל מע"מ:	מחיר יח'	מטבע - שער סה"כ שורה	מטבע	מחיר יח'	מטבע	מחיר יח'	מטבע	מחיר יח'	מטבע	מחיר יח'	מטבע
פריט בש"ח	100.00	100.00	100.00	1.000	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
פריט ביורו	200.00	200.00	200.00	5.000	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00
<p>סה"כ ביניים: 2,100.00</p>												

איור 1: שורות החשבונית בכספית

החשבונית בכספית

Invoice 10029		דוגמא		23/06/2011	
Description	Qty	Item Price	Exch. Rate	Ext. Price	
פריט בש"ח	1	100.00	1	100.00	
פריט ביורו	2	200.00	5	2000.00	
Sub Total:				2,100.00	
Total:				2,100.00	
Vat 16%:				336.00	
Grand Total:				2,436.00	

איור 2: הדפסה באנגלית עם תרגום לש"ח

כל הזכויות שמורות © כספית תוכנה בע"מ

ה-API מציית להגדרות שנקבעו בכספית.

דוגמה לקוד VB להדפסה בעברית

```
Call objCaspit.PrintInvoice (CLng (strInvoiceNum) )
```

דוגמה לקוד VB להדפסה באנגלית

```
Call objCaspit.PrintInvoice (CLng (strInvoiceNum) ,  
caspitAPI.Languages.English)
```

קבלות

יצירה של קבלה – CreateReceipt

```
Function CreateReceipt(iCustomersID As Long, datReceiptDate As Date, curTotal As Currency, curTax As Currency, curNetPmt As Currency, strDetails As String, strComments As String, strInvoiceNumbers As String, varReceiptLineItems, [strCurrency As String = "NIS"], [curRate As Currency = 1]) As Long
```

המתודה מחזירה את מספר הקבלה. בכספית יש אפשרות לסוג קבלות (בדומה לסיווג תנועה של חשבוניות). המתודה לא מאפשרת לשלוח סיווג תנועה. ה-API יוצר את הקבלה תמיד עם סיווג תנועה 2-קבלה.

הפרמטרים הם:

- ❖ iCustomersID – ה-ID של הלקוח עבורו יוצרים את הקבלה.
- ❖ datReceiptDate – תאריך הקבלה
- ❖ curTotal – סה"כ התשלום בקבלה. הסכום צריך להיות שווה לסכום התשלום בפועל + הניכוי במקור.
- ❖ curNetPmt – התשלום שהתקבל בפועל (לאחר הניכוי במקור). צריך להיות שווה לסכום של התשלום בשורות הקבלה.
- ❖ curTax – סכום הניכוי במקור.
- ❖ strDetails – פרטי הקבלה. הפרטים אינם מודפסים ללקוח. (3.2)
- ❖ strComments – תיאור המוצר או השירות. התיאור מודפס ללקוח. (3.2)
- ❖ strInvoiceNumbers – רשימה של חשבוניות שהקבלה "סוגרת". המערכת אינה בודקת שמספרי החשבוניות חוקיים. אם יש יותר מחשבונית אחת הפרד את מספר החשבוניות בפסיק.
- ❖ varReceiptLineItems – מערך של שורות קבלה. מסוג ReceiptLineItems
- ❖ strCurrency ו-curRate – מאפשר ליצור קבלה במט"ח.

שורת קבלה – ReceiptLineItem

האובייקט מייצג שורת קבלה. יש לו התכונות הבאות:

AccountNumber
Amount
Bank
Branch
Check
CheckDate
PaymentTypesID

- ❖ AccountNumber – עבור שיק: מספר החשבון, עבור כ"א: מספר הכרטיס.
- ❖ Amount – סכום התשלום הנומינלי. המטבע והשווי בש"ח נקבעים ברמת הקבלה על-ידי strCurrency ו-curRate.
- ❖ Bank – עבור שיק: מספר הבנק. רשימת הבנקים והמספר שלהם מפורטת בנספח א'. זו רשימה סטנדרטית של בנקים בישראל ומופצת ע"י בנק ישראל. אם התשלום הוא במזומן שם את הערך 0 בתכונה זו.
- ❖ Branch – עבור שיק: מספר הסניף.
- ❖ Check – עבור שיק: מספר השיק, עבור כ"א: מספר השובר.

כל הזכויות שמורות © כספית תוכנה בע"מ

❖ CheckDate – עבור שיק: תאריך הפרעון של השיק, עבור כ"א: תאריך תוקף (לדוגמה: 06/2001) ←
(01/06/2011)

❖ PaymentTypesID – ה-ID של סוג התשלום (מזומן, כ"א, העברה בנקאית וכו'). רשימת סוגי התשלום מפורטת בנספח ב'.

יצירה של קבלה על-ידי XML – CreateReceiptXML

```
Function CreateReceiptXML(strXML as String) as Long
```

ראה פירוט של מבנה ה-XML בפרק המתאים.

הפונקציה מחזירה את מספר הקבלה שנוצרה.

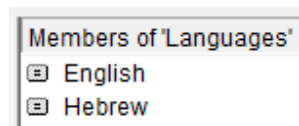
הדפסה של קבלה

המתודה מדפיסה קבלה בעברית או באנגלית

```
Public Sub PrintReceipt(iReceiptNumber As Long, Optional Lang As  
Languages = Hebrew)
```

❖ מספר הקבלה (לדוגמה 30028) -- iReceiptNumber

❖ Lang – מספר השפה. רצוי להשתמש בקבועים שמוגדרים באנומריצה Languages.



ראה הסבר בהקדמה על מספר העותקים שיודפסו עם המקור.

חשבונית/קבלה

יצירה של חש/קבלה

```
Function CreateInvoiceKabala(iCustomersID As Long, datInvoiceDate As Date, strComments As String, fltRebatePercent As Double, curRebateAmount As Currency, blnRoundTotal As Boolean, fltVAT As Double, blnChargeVAT As Boolean, strInvoiceDetails As String, iTrxCodesID As Long, varLineItems, curTotal As Currency, curTax As Currency, curNetPmt As Currency, varReceiptLineItems) As Long
```

הפרמטרים דומים לפרמטרים של חשבונית בשילוב עם הפרמטרים של יצירת קבלה. המתודה מחזירה את מספר (לדוגמה 20004, לא ה-ID) של החש/קבלה שנוצרה.

יצירה של חש/קבלה ב-XML

```
Public Function CreateInvoiceKabalaXML(strXML As String) As Long
```

ראה פירוט של מבנה ה-XML בפרק המתאים.

המתודה מחזירה את מספר (לדוגמה 20004, לא ה-ID) של החש/קבלה שנוצרה.

הצגה של חש/קבלה

```
Function DisplayInvoiceKabala(iInvoiceNumber As Long)
```

ה-API יפתח את כספית ויציג את החש/קבלה המבוקשת.

❖ iInvoiceNumber -- מספר החשבונית (לדוגמה 20028).

הדפסה של חש/קבלה

```
Sub PrintInvoiceKabala(iInvoiceNumber As Long)
```

❖ iInvoiceNumber -- מספר החשבונית (לדוגמה 20028).

ראה הסבר בהקדמה על מספר העותקים שיודפסו עם המקור.

שליפה של חש/קבלה

```
Function GetInvoiceKabalaByNumber(iInvoiceNumber As Long) As CaspitAPI.InvoiceKabala
```

❖ iInvoiceNumber -- מספר החשבונית (לדוגמה 20028).

שליפה של כל החש/קבלה במערך

```
Function GetInvoicesKabala() As InvoiceKabala()
```


חשבונית עסקה

יצירה של חשבונית עסקה

```
Function CreateInvoiceIska(iCustomersID As Long, datInvoiceDate As Date,  
strComments As String, fltRebatePercent As Double, curRebateAmount As  
Currency, blnRoundTotal As Boolean, fltVAT As Double, blnChargeVAT As  
Boolean, strInvoiceDetails As String, iTrxCodesID As Long, iInvoiceType  
As InvoiceTypes, varLineItems, [datDueDate As Date]) As Long
```

הצגה של חשבונית עסקה בכספית

```
Function DisplayInvoiceIska(iInvoiceNumber As Long)
```

הדפסה של חשבונית עסקה

```
Sub PrintInvoiceIska(iInvoiceNumber As Long)
```

ראה הסבר בהקדמה על מספר העותקים שיודפסו עם המקור.

חשבונית זיכוי

יצירה של חשבונית זיכוי

```
Function CreateInvoiceZikui(iCustomersID As Long, datInvoiceDate As Date, strComments As String, fltRebatePercent As Double, curRebateAmount As Currency, blnRoundTotal As Boolean, blnChargeVat As Boolean, strInvoiceDetails As String, iTrxCodesID As Long, varLineItems, Optional strInvoices as string) As Long
```

בכספית הסכומים בשורות של חשבונית זיכוי הם במינוס. ה-API בנוי כך שגם אם הסכומים חיוביים ה-API יתן להם סימן מינוס.

לחשבונית זיכוי אין תאריך תשלום.

strInvoices = מספר בודד או רשימה של מספרי חשבוניות מס וחש/קבלה שחשבונית הזיכוי קשורה אליהם. אם אתה שולח יותר ממספר אחד אז הפרד את המספרים בפסיקים. לדוגמה: "10023" או "10023,20901,10090".

כיצד ה-API ידע האם מדובר במספר של חשבונית מס או חשבונית/קבלה?

תשובה:

ה-API לא ידע. אם לדוגמה תשלח "104" ובכספית קיימים גם חשבונית מס מספר 104 וגם חשבונית/קבלה מספר 104 אז ה-API יעדכן את שניהם ויקשור את שניהם לחש/זיכוי החדשה.

זו אחת הסיבות שחשוב מאד להפריד בין הסדרות של מספרי המסמכים בכספית.

חשוב לקשר את חשבונית הזיכוי לחשבונית מס או לחש/קבלה בכדי לעזור לכספית לחשב נכון את יתרת הלקוחות.

המתודה מחזירה את מספר (לדוגמה 50004, לא ה-ID) של החש/זיכוי שנוצרה.

יצירה של חשבונית זיכוי ב-XML

```
Function CreateInvoiceZikuiXML(strXML as string) as long
```

ראה פירוט של מבנה ה-XML בפרק המתאים.

המתודה מחזירה את מספר (לדוגמה 50004, לא ה-ID) של החש/זיכוי שנוצרה.

הצגה של חשבונית זכוי בכספית

```
Function DisplayInvoiceZikui(iInvoiceNumber As Long)
```

הדפסה של חשבונית זכוי

```
Sub PrintInvoiceZikui(iInvoiceNumber As Long)
```

ראה הסבר בהקדמה על מספר העותקים שיודפסו עם המקור.

תעודות החזר

יצירה של תעודת החזר – CreateRefund

תעודת החזר היא מסמך המתעד החזר של תשלום ללקוח.

```
Function CreateRefund(iCustomersID As Long, _  
    datRefundDate As Date, _  
    curTotal As Currency, _  
    strDetails As String, _  
    strComments As String, _  
    varReceiptLineItems As Variant) As Long
```

המתודה מחזירה את מספר תעודת החזר.

הפרמטרים הם:

- ❖ iCustomersID – ה-ID של הלקוח עבורו יוצרים את תעודת החזר.
- ❖ datRefundDate – תאריך התעודה.
- ❖ curTotal – סה"כ התשלום בקבלה. הסכום צריך להיות שווה לסכום שורות התשלומים.
- ❖ strDetails – פרטי הקבלה. הפרטים אינם מודפסים ללקוח.
- ❖ strComments – תיאור המוצר או השירות, או אולי תיאור סיבת החזר. הפרטים מודפסים ללקוח.
- ❖ varReceiptLineItems – מערך של שורות קבלה. מסוג ReceiptLineItems

הערה:

בכספית הסכומים בשורות של תעודת החזר הם במינוס, וגם curTotal במינוס. ה-API בנוי כך שגם אם הסכומים חיוביים ה-API יתן להם סימן מינוס.

הערה חשובה:

שם לב שאם העסקה בוטלה, בדר"כ לא די בהפקת תעודת החזר. עליך גם להפיק חשבונית זיכוי בכדי לקבל חזרה את המע"מ ששילמת, ולחייב את הלקוח לשלם את המע"מ שהוא התקזז עליו.

יצירה של תעודת החזר על-ידי XML – CreateRefundXML

```
Function CreateRefundXML(strXML as String) as Long
```

ראה פירוט של מבנה ה-XML בפרק המתאים.

הפונקציה מחזירה את מספר תעודת החזר שנוצרה.

הצגה של תעודת החזר בכספית

```
Function DisplayRefund(iReceiptNumber As Long)
```

הדפסה של תעודת החזר

```
Sub PrintRefund(iRefundNumber As Long)
```

ראה הסבר בקדמה על מספר העותקים שיודפסו עם המקור.

חיובי זמן

ה-API מאפשר יצירה של חיובי זמן (שנמצאים בשימוש בכספית+יועצים).

יצירת חיוב זמן

```
Function CreateTimeBill(iCustomersID As Long, datStart As Date, datEnd As Date, strDescription As String, strAppointmentID As String, iInvoiceNumber As String, [curRate As Currency], [strCurrency As String]) As Long
```

- חיוב הזמן הוא מתאריך datStart (יום ושעה) עד לתאריך datEnd (יום ושעה).
- ניתן לשייך חשבונית לחיוב. אין בדיקה ב-API שהחשבונית אכן קיימת בכספית.
- ניתן לקבוע לחיוב תעריף לשעה ומטבע לחיוב.
- ניתן לשייך לחיוב Appointment מתוכנה צד ג' (לדוגמה Outlook). מתאים למקרה שבו צריך לעדכן את החיובים כתוצאה משינוי לו"ז בתוכנה צד ג'.
- אם ה-strAppointmentID שונה מ "" (מחרוזת ריקה) אז כספית תמחק את חיוב הזמן שקיים מול הפגישה הנתונה לפני שהיא תיצור חיוב חדש.

דוגמה מ-API Demo

טופס יצירת חיובי זמן

שם הלקוח	תאריך התחלה	סיום	ש"ע	חיוב לשעה	מטבע	תיאור הפגישה
Mr. Li Huyi - 40	25/03/11 14:27	25/03/11 15:57	1.5	200	NIS	פגשת עבודה
Mr. Li Huyi - 40	25/03/11 19:27	25/03/11 21:27	2	300	NIS	יעוץ עסקי
רפאל ציוני - 42	26/03/11 12:27	26/03/11 14:46	2.3	400	USD	הכנת תוכנית עבודה

ובכספית:

חיוב זמן							
עסק מספר 123456789 2							
א	לקוח	איש קשר	תאריך	שעות	תעריף	מט'	חיוב
	Mr. Li Huyi		25/03/11	1.50	200.00	ש	300.00
	Mr. Li Huyi		25/03/11	2.00	300.00	ש	600.00
	רפאל ציוני		26/03/11	2.32	400.00	\$	3,294.30

מחירוניה

המחירוניה של כספית (רלוונטיים בגרסת כספית+מלאי) מיוצגים על-ידי האוביקט PriceList:

Members of 'PriceList'

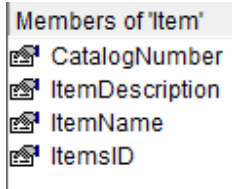
- Comments
- ListID
- ListNum
- ProfitPercent

למחירון התכונות הבאות:

- ListID – ה-ID בטבלת מחירוניה.
- ListNum – מספר המחירון (1, 2, 3, ...)
- ProfitPercent – אחוז הרווח במחירון (0 = מחירון שרירותי. כלומר המשתמש מכניס את מחירי המכירה בעצמו, 10 = אחוז הרווח הוא 10% וכו').
- Comments = תיאור המחירון (לדוגמה: "מחירון מספר 1")

פריטים

הפריטים של כספית (רלוונטיים בגרסת כספית+מלאי) מיוצגים על-ידי האובייקט Item:



לפריט התכונות הבאות:

- iltemsID – ה-ID בטבלת Items.
- ItemName – שם הפריט
- ItemDescription – תיאור הפריט
- CatalogNumber – מספר קטלוגי של הפריט.

הערה: בגרסה הנוכחית של ה-API עדיין לא ניתן לשלוף את המחירים של הפריט במחירונים השונים.

פעולות נוספות

הרצה של כספית

```
Function RunCasplit()
```

המתודה תריץ את כספית.

שליפה של אחוז המע"מ

```
Function GetVatRate([datDate As Date]) As Double
```

המודה תחזיר את אחוז המע"מ ביום הרלוונטי.

הערך החוזר הוא באחוזים (כלומר 16.5 כאשר אחוז המע"מ הוא 16.5%).

אם התאריך חסר כספית תחזיר את אחוז המע"מ ליום הנוכחי.

שליפה של ערך Registry של כספית

```
Public Function GetCasplitRegSetting(EntryName As String, ByVal  
varDefault As Variant) As Variant
```

כספית שומרת ערכים שונים ב-Registry תחת HKEY_LOCAL_MACHINE/SOFTWARE/CASPLIT. אתה יכול לשלוף ערכים משם באמצעות המתודה. לדוגמה שליפה של שמו של קובץ הגיבוי האחרון:

```
objCasplit.GetCasplitRegSetting("LastBackupDate ", "")
```

שליפה של ערך מטבלת הפרמטרים של כספית

```
Public Function GetSystemParameter(strParameterName As String) As  
Variant
```

כספית מחזיקה ערכי מערכת שונים בטבלת פרמטרים. אם דרוש לך ערך מסוים אתה יכול לשלוף אותו בעזרת המתודה. ערכים לדוגמה:

sysSystemParameters		
vcParameterDescription	vcParameterValue	vcParameterName
התאריך של עדכון שערי החליפין	11/08/2011 13:52	datLastCurrenciesUpdate
התאריך של הפילטר האחרון של עדכון tblCustomerBalance	31/12/2011	qryCustomerBalanceDate
המסלול לקובץ הגיבוי	backup	strBackupPath

שינוי של המדפסת של כספית

```
Sub SetCasplitPrinter(strPrinterName As String)
```

ניתן לשנות את המדפסת של כספית למדפסת הרצויה.

משלוח של מסמך בדואר אלקטרוני

```
Sub SendDocByMail(iDocNumber As Long, iObjectType As ObjectType,  
strSubject As String, strForm As String, strFromName As String,  
strRecipients As String, strRecipientsCC As String, strRecipientsBCC  
As String, strHTMLBody As String)
```

המתודה מאפשרת לשלוח את המסמך הרצוי במייל.
המתודה משתמשת באובייקט Dimac.Jmail (ראה בהקדמה) שחייב להיות מותקן על המחשב.

לפני הקריאה למתודה צריך להגדיר את שרת המייל, ואת פרטי החשבון (MailServer, MailServerPassword, MailServerUserName

- iObjectType – סוג המסמך (חשבונית, קבלה וכו').
הערכים מפורטים ב-Public Enum ObjectType בנוסף סוגי המסמכים מפורטים בנספחים.
- strForm – לדוגמה dror@caspit.biz
- strFromName – השם הידיוטי. לדוגמה: Caspit Software.
- strReceipients – למי לשלוח את המייל. אפשר לשלוח מספר כתובת מייל מופרדות בפסיקים. לדוגמה:
dror@caspit.biz, info@caspit.biz, support@caspit.biz

משלוח מייל

```
Sub SendMail(strSubject As String, strForm As String, strFromName As String, strReceipients As String, strReceipientsCC As String, strReceipientsBCC As String, strHTMLBody As String, [strAttachmentFileName]) As String
```

המתודה מאפשרת לשלוח מייל. היא נוספה למען נוחות השימוש ולא שייכת ממש לכספית.

לפני הקריאה למתודה צריך להגדיר את שרת המייל, ואת פרטי החשבון בשרת: MailServer, MailServerPassword, MailServerUserName

בכדי לשלוח את המייל, ה-API משתמש באובייקט Dimac.jmail. לכן חייבים [להתקין גם את Dimac w3Jmail](#) (ההתקנה חיינית).

ביצוע גיבוי

```
Function PerformBackup() as String
```

ה-API מאפשר לך להורות לכספית לבצע גיבוי. כספית תבצע גיבוי לפי מה שנקבע בפרטי העסק, כלומר המסלול של הגיבוי, והאם הגיבוי הוא בפורמט כספית או כספיות (3.2) נקבע לפי מה שהמשתמש הגדיר בכספית.

המתודה מחזירה את שם הקובץ אליו בוצע הגיבוי.

הגיבוי הוא פעולה מורכבת ויתכנו תקלות שונות. ה-API מאפשר לך להגדיר Timeout (בשניות) שהוא יחכה לוסף הגיבוי. אם הגיבוי לא יסתיים עד אז, ה-API "זרוק" שגיאה.

ברירת מחדל: 60 שניות.

```
50 Set objCaspit = New CaspitAPI.Caspit
60 With objCaspit
70     .CaspitPath = "C:\Caspit\Caspit30.mde"
80     .DebugMode = False
85     .BusinessID = 2
90     .BackupTimeOutSeconds = 100 'Seconds
100    strLastBackupFile = .PerformBackup()
110 End With
120 Set objCaspit = Nothing
```


הוסף הודעת שגיאה לטבלת שגיאות של כספית

Sub AddErrorToCasplit(strErrorMessage As String)

המתודה מאפשרת להוסיף שגיאה ללוג השגיאות של כספית. את הלוג ניתן לראות דרך דו"ח שגיאות מהמסך הראשי של כספית.

תיאור של ה-XML

פרק זה מתאר את ה-XML שאותו אפשר לשלוח ל-API. זהו גם ה-XML שה-API מחזיר.

עדכון ערכים, שדות ריקים ו-NULL

בכל API קיימת בעיה בעת עדכון של נתונים: כיצד להתייחס לשדות ריקים? האם הכוונה היא שהערך החדש של השדה הוא ריק או שהכוונה שהערך של השדה אינו ידוע (ולכן אין לשנות את הערך הקיים). עבור שדות מסוג טקסט ה-API של כספית עובד כך:

1. אם השדה אינו מופיע ב-XML או שהשדה כן מופיע אבל הערך ריק – ה-API לא יעדכן את השדה.

2. אם הערך של השדה הוא NULL אז ה-API יאפס את השדה.

3. אם הערך של השדה אינו ריק וגם שונה מ-NULL אז ה-API יעדכן את השדה.

עבור שדות מסוג מספרי (לדוגמה, Charge) – לא ניתן לעדכן את השדה ל-NULL. אם תשלח ב-XML את הערך NULL ה-API יחזיר שגיאה. אם לא תשלח את השדה בכלל, או אם תשלח את הערך 0, אז ה-API יתעלם מהערך ששלחת.

לדוגמה, בעת עדכון לקוח (זה ה-XML היחיד שנשלח בגרסה זו עבור עדכון, אבל יתכן שבעתיד נרחיב את היכולות של ה-API):

דוגמה 1

```
<Caspit>
<Customer>
<CustomersID>213</CustomersID>
</Customer>
</Caspit>
```

במקרה זה לא יתבצע שום שינוי בשדות של הלקוח (משום שהשדה CustomersID משמש רק לזיהוי של הלקוח שאותו יש לעדכן ואין שום שדה אחר ב-XML).

דוגמה 2

```
<Caspit>
<Customer>
<CustomersID>213</CustomersID>
<Email></Email>
</Customer>
</Caspit>
```

גם במקרה זה לא יתבצע שום שינוי בשדות של הלקוח.

דוגמה 3

```
<Caspit>
<Customer>
<CustomersID>213</CustomersID>
<Email>NULL</Email>
</Customer>
</Caspit>
```

במקרה זה ה-API "יאפס" את הערך של השדה כתובת המייל.

דוגמה 4

```
<Caspit>
<Customer>
<CustomersID>213</CustomersID>
<Email>dror@caspit.biz</Email>
```

```
</Customer>
</Casplit>
```

במקרה זה ה-API ישנה את הערך של השדה כתובת המייל ל-dror@casplit.biz

XML עבור יצירת לקוח ועדכון

להלן ה-XML עבור שליפה של לקוחות מכספית. בסוגריים [ברירת המחדל בעת יצירת לקוח אם לא תשלח ערך]:

```
<Casplit>
<Customer>
  <CustomersID /> → משמש לזיהוי הלקוח. יוצרים לקוח. מעדכון.
  <CustomerNumber /> → (3.2) מספר הלקוח [אם תשלח 0 אז כספית תיתן מספר לקוח גדול ב-1 ממספר הלקוח הגדול ביותר. אם תשלח מספר לקוח שכבר קיים, אז ה-API יחזיר שגיאה (שמקורה באינדקס על השדה) על כפילות בערכים.]
  <OsekMorshe />
  <Address1 />
  <Address2 />
  <BusinessName />
  <City />
  <PostalCode />
  <ContactFirstName />
  <ContactLastName />
  <Email />
  <CreditCard /> → (הוסר ב 3.2)
  <ExpDate /> → (הוסר ב 3.2)
  <Phone1 />
  <Phone2 />
  <Fax />
  <OpenBalance /> → [0] יתרת פתיחה של הלקוח
  <Comments />
  <ForeginKey /> → מפתח זר
  <BankAccountNumber />
  <BankBranch /> → סניף הבנק. יכול להיות שם הסניף או מספרו
  <BanksID /> → [0] מספר הבנק לפי רשימת הבנקים שבנספחים
  <PaymentTerms /> → [שוטף=1] PaymentTerms הערך הוא לפי הקבועים של
  <PaymentDays /> → [0] מספר הימים עבור תנאי תשלום שוטף+
  <Charge /> → [0] חיוב של שעת עבודה (עבור כספית+יועצים)
  <ChargeCurrency /> → Charge המטבע עבור שדה [NIS]
  <PriceListNum /> → [1] מספר המחירון
  <Extra1></Extra1> → שדה הערות 2 (3.2)
  <Extra2></Extra2> → שדה הערות 3 (3.2)
</Customer>
</Casplit>
```

XML בסיסי עבור חשבוניות מס, חשבוניות עסקה, חשבוניות זיכוי

הרעיון הבסיסי הוא שכספית מייצרת מסמכים Document. המסמך שיוצר תלוי במתודה שבה נשתמש.

ה-XML שעליו מבוסס ה-API נראה כך:

```
<Casplit>
<Document>
  <DocDate></DocDate>
  <CustomerID></CustomerID> or
  <CustomerForeignKey></CustomerForeignKey>
  <DocComments></DocComments>
```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

<DocDetails></DocDetails>
<RebatePercent></RebatePercent>
<RebateAmount></RebateAmount>
<RoundTotal></RoundTotal>
<VAT></VAT> → אם הערך חסר או 0 אז כספית תשתמש בערך המע"מ הנוכחי
<ChargeVAT></ChargeVAT>
<TrxCodesID></TrxCodesID>
<DueDate></DueDate> → רק לחשבונית
<LineItems>
  <LineItem number="1">
    <ItemDescription></ItemDescription>
    <ItemQuantity></ItemQuantity>
    <UnitPrice></UnitPrice>
    <Currency></Currency> → NIS: אם הערך חסר המערכת תציב
    <CurrencyRate></CurrencyRate> → 1.0: אם הערך חסר המערכת תציב
  </LineItem>
</LineItems>
</ Document >
</Caspit>

```

ערכים

תאריכים

הפורמט המלא של תאריכים הוא: 2003-08-22T13:45:55.9821000+02:00

ה-API יתרגם את התאריך לתאריך הבא: 13:45:55 22/8/2003. כלומר, הוא מתעלם מחלקי השניה, ומאיזור הזמן. לרוב הצרכים של ה-API מספיק להעביר את חלק התאריך בלבד (ללא הזמן): 2003-08-22.

מע"מ

ניתן לשלוח לכספית 0 בתור ערך המע"מ. במקרה זה כספית תשתמש בערך המע"מ כפי שהוא מוגדר בתוכנה. אם תשלח ערך שונה מ-0 כספית תשתמש בו לחלק מן החישובים, ובחלק אחר מן החישובים תשתמש בערך המע"מ המוגדר בתוכנה. זו מגבלה של המבנה של כספית ואין לה פיתרון בשלב זה. המטרה של האפשרות שניתנה למתכנת לשלוח את ערך המע"מ היא לעקוף קוד ב-API שמבצע גישה לכספית לשליפה של ערך המע"מ. קוד זה גרם לבעיות במספר התקנות. הכוונה איננה לאפשר למתכנת שליטה על ערך המע"מ שכספית משתמשת בו, לכן צריך תמיד לשלוח את ערך המע"מ העדכני.

מטבעות

בשדה Currency תחת Lineltem ניתן לשלוח מטבע.

ה-API תומך במטבעות הבאים: NIS, GBP, USD, EUR, JPY, NZD, AUD, CAD.

אם השדה חסר או ריק ה-API יניח שמדובר בש"ח. אם שולחים מטבע יש לשלוח גם את שער החליפין, CurrencyRate. כלומר, ה-API אינו שולף את ערך החליפין בעצמו, אלא לוקח אותו מה-XML שהוא מקבל.

אם ה-API מקבל מטבע שאינו מוכר לו הוא מחזיר שגיאה.

דוגמא – יצירת חשבונית

דוגמה עבור יצירת חשבונית:

```

<Caspit>
<Document type="Invoice">
<CustomerID>389</CustomerID>
<TrxCodesID>13</TrxCodesID>
<DocDate>2008-05-18</DocDate>

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

<DocComments></DocComments>
<DocDetails></DocDetails>
<RebatePercent>0</RebatePercent>
<RebateAmount>0.00</RebateAmount>
<RoundTotal>False</RoundTotal>
<ChargeVAT>True</ChargeVAT>
<VAT>15.5</VAT>
<LineItems>
  <LineItem number="1">
    <ItemDescription>p1</ItemDescription>
    <ItemQuantity>1.1</ItemQuantity>
    <UnitPrice>1.98</UnitPrice>
  </LineItem>
  <LineItem number="2">
    <ItemDescription>p2</ItemDescription>
    <ItemQuantity>2.2</ItemQuantity>
    <UnitPrice>2.00</UnitPrice>
    <Currency>USD</Currency>
    <CurrencyRate>3.45</CurrencyRate>
  </LineItem>
  <LineItem number="3">
    <ItemDescription>p3</ItemDescription>
    <ItemQuantity>3.39</ItemQuantity>
    <UnitPrice>3.568</UnitPrice>
    <Currency>GBP</Currency>
    <CurrencyRate>7.0932</CurrencyRate>
    <ChargeVAT>True</ChargeVAT>
  </LineItem>
</LineItems>
</Document>
</Caspit>

```

UnitPrice -- מחיר יחידה. אם אתה מחליץ את מחיר היחידה מהמחיר כולל מע"מ, רצוי מאד לשלוח את מחיר היחידה עם 4 ספרות אחרי הנקודה לכל הפחות. כספית תמיג את מחיר היחידה מעוגל ל-2 ספרות (זו ברירת המחדל, המשתמש יכול לקבוע את מספר הספרות בעצמו), אבל תשתמש בכל הספרות לחישוב המע"מ. זה ימנע מצב אופייני שבו המחיר כולל מע"מ שונה באגורה מהמחיר המקורי שממנו יצאת

Currency – המטבע. האפשרויות הן:

- NIS – ש"ח (ברירת מחדל)
- USD – דולר ארה"ב
- GBP – לירה שטרלינג
- EUR – יורו
- JPY – י"ן יפני
- NZD – דולר ניו זילנדי
- AUD – דולר אוסטרלי
- CAD – דולר קנדי

XML עבור יצירת קבלה

ראה הסבר על משמעות השדות בפרמטרים של המתודה CreateReceipt.

```

<Caspit>
  <Document type="receipt">

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

<CustomersID></CustomersID>
<DocDate></DocDate>
<Total></Total>
<Tax></Tax>
<NetPmt></NetPmt>
<Details></Details> → פרטים שאינם מוצגים ללקוח (3.2)
<Comments></Comments> → תיאור השירות/המוצר, מודפס ומוצג (3.2)
<InvoiceNumbers></InvoiceNumbers>
<ReceiptLineItems>
  <ReceiptLineItem number="1">
    <AccountNumber></AccountNumber>
    <Amount></Amount>
    <Bank></Bank> → אם הערך חסר או 0 אז כספית תשאר
    <Branch></Branch> → אם הערך חסר או 0 אז כספית
    <Check></Check>
    <CheckDate></CheckDate>
    <PaymentTypesID></PaymentTypesID>
  </ReceiptLineItem>
</ReceiptLineItems>
</Document>
</Caspit>

```

את השדה ריק

תשאיר את השדה ריק

XML עבור יצירת תעודת החזר

ראה הסבר על משמעות השדות בפרמטרים של המתודה CreateRefund.

שם לב:

יש לשלוח את הסכומים **במינוס**.

ה-**Total** יכול להיות 0. ה-API מחשב אותו כסכום השורות

```

<Caspit>
  <Document type="Refund">
    <CustomerID>-30100111</CustomerID>
    <DocDate>2011-08-11</DocDate>
    <Total>0</Total>
    <Details></Details>
    <Details></Details> → פרטים שאינם מוצגים ללקוח (3.2)
    <Comments></Comments> → תיאור השירות/המוצר, מודפס ומוצג (3.2)
    <ReceiptLineItems>
      <ReceiptLineItem number="1">
        <AccountNumber></AccountNumber>
        <Amount>-111.11</Amount>
        <Bank></Bank>
        <Branch></Branch>
        <Check></Check>
        <CheckDate></CheckDate>
        <PaymentTypesID>1</PaymentTypesID>
      </ReceiptLineItem>
      <ReceiptLineItem number="2">
        <AccountNumber>6262677</AccountNumber>
        <Amount>-222.33</Amount>

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

        <Bank>1</Bank>
        <Branch>7278</Branch>
        <Check>100</Check>
        <CheckDate>2011-10-23</CheckDate>
        <PaymentTypesID>2</PaymentTypesID>
    </ReceiptLineItem>
    <ReceiptLineItem number="3">
        <AccountNumber>5353656</AccountNumber>
        <Amount>-333.33</Amount>
        <Bank>1</Bank>
        <Branch></Branch>
        <Check></Check>
        <CheckDate></CheckDate>
        <PaymentTypesID>6</PaymentTypesID>
    </ReceiptLineItem>
</ReceiptLineItems>
</Document>
</Caspit>

```

דוגמא – יצירת קבלה

```

<Caspit>
  <Document type="Receipt">
    <CustomerID>389</CustomerID>
    <DocDate>2008-05-24</DocDate>
    <Total>665.28</Total>
    <Tax>51.39</Tax>
    <NetPmt>613.89</NetPmt>
    <Details>לקוח קטנוני ומעצבן</Details>
    <Comments>אירוח מושלם ליום ההולדת </Comments >
    <InvoiceNumbers>20001,20002</InvoiceNumbers>
    <Currency>USD</Currency> → Default: NIS
    <Rate>3.8165</Rate> → Default: 1.000
    <ReceiptLineItems>
      <ReceiptLineItem number="1">
        <AccountNumber>2321321</AccountNumber>
        <Amount>513.89</Amount>
        <Bank>8</Bank>
        <Branch>566</Branch>
        <Check>5455</Check>
        <CheckDate>2008-05-24</CheckDate>
        <PaymentTypesID>2</PaymentTypesID>
      </ReceiptLineItem>
    </ReceiptLineItems>
    <ReceiptLineItems>
      <ReceiptLineItem number="2">
        <AccountNumber></AccountNumber>
        <Amount>100.00</Amount>
        <Bank>0</Bank>
        <Branch>0</Branch>
        <Check></Check>
        <CheckDate></CheckDate>
        <PaymentTypesID>1</PaymentTypesID>
      </ReceiptLineItem>
    </ReceiptLineItems>
  </Document>
</Caspit>

```

```

</ReceiptLineItems>

</Document>
</Caspit>

```

XML עבור שליפת חשבונית

```

<Caspit>
<Document type="Invoice"> → ראה רשימת סוגי מסמכים בהמשך
  <CustomersID>369</CustomersID>
  <BusinessName>סדן דרוור</BusinessName>
  <CustomerOsekMorshe></CustomerOsekMorshe>
  <Address1>éì"â 8</Address1>
  <Address2>ãéøä 4</Address2>
  <City>ú"à</City>
  <PostalCode>91919</PostalCode>
  <ContactDetails>åøëä ùââié</ContactDetails>
  <Phone1></Phone1>
  <Phone2></Phone2> (3.2)
  <Fax></Fax>
  <TrxCodesID>13</TrxCodesID> → ראה רשימת סווגי תנועות בנספחים
  <DocDate>2008-06-05T00:00:00</DocDate>
  <DocComments></DocComments>
  <DocDetails></DocDetails>
  <Total1>128</Total1> → סה"כ של הפריטים, לפני הנחה מסחרית, הנחת עיגול וחע"ח
  <RebatePercent>10</RebatePercent>
  <RebateAmount>12.8</RebateAmount>
  <RoundTotal>True</RoundTotal> → האם לעגל את המחיר לשקל שלם
  <RebateRoundAmount>.0485</RebateRoundAmount> → הנחת עיגול
  <TotalBeforeVat>115.1515</TotalBeforeVat>
  <ChargeVAT>True</ChargeVAT>
  <VAT>15.5</VAT> → אחוז החע"ח (באחוזים)
  <VATAmount>17.8485</VATAmount>
  <GrandTotal>133</GrandTotal>
  <StatusID>2</StatusID> → ראה טבלת סטטוסים
  <ReceiptNumbers>20001,20012</ReceiptNumbers> → מספרי הקבלות שהונפקו כנגד החשבונית
  <DueDate>2008-07-31T00:00:00</DueDate> → מועד התשלום
  <LineItems>
    <LineItem number="1">
      <ItemDescription>ðñé òø (î')</ItemDescription>
      <ItemQuantity>20</ItemQuantity>
      <UnitPrice>2.9</UnitPrice> → מחיר לפני חע"ח
      <Currency>₪</Currency>
      <CurrencyRate>1</CurrencyRate> → שער חליפין
      <ExtendedPrice>58</ExtendedPrice> → סה"כ השורה
    </LineItem>
    <LineItem number="2">
      <ItemDescription>ðéðåú îñâøåú (éç')</ItemDescription>
      <ItemQuantity>100</ItemQuantity>
      <UnitPrice>.7</UnitPrice>
      <Currency>₪</Currency><CurrencyRate>1</CurrencyRate>
      <ExtendedPrice>70</ExtendedPrice>
    </LineItem>
  </LineItems>
</Caspit>

```



```
</Document>
</Caspit>
```

הערות:

1. כספית מחזירה הן את ה-ID של הלקוח והן את פרטי הכתובת ואיש הקשר שלו. הסיבה היא שבכספית אפשר לשנות את הפרטים ברמת החשבונית, בכדי לתמוך במשלוח חשבונית לכתובת שונה מהכתובת שברשומת הלקוח.
2. בתג Document ישנו attribute בשם type. הוא יכול להיות אחד מהערכים הבאים:

- ❖ Invoice - חשבונית מס
- ❖ InvoiceKabala - חשבונית מס/קבלה
- ❖ Proforma - חשבונית עסקה
- ❖ InvoiceZikui - חשבונית זיכוי
- ❖ Offer - הצעת מחיר
- ❖ Order - הזמנת עבודה
- ❖ Receipt - קבלה
- ❖ Refund - תעודת החזר

XML עבור רשימת לקוחות

דוגמה עבור רשימה של לקוחות כפי שכספית מחזירה אותם:

```
<Caspit>
  <Customer>
    <CustomersID>389</CustomersID>
    <BusinessName>ëääé</BusinessName>
    <OsekMorshe>656575778</OsekMorshe>
    <Address1>äiç"é 7</Address1>
    <Address2>÷âïä á'</Address2>
    <City>øàù äðéi</City>
    <PostalCode>82829</PostalCode>
    <ContactFirstName>îüä</ContactFirstName>
    <ContactLastName>ëääé</ContactLastName>
    <Phone1>09-1234567</Phone1>
    <Phone2>052-2345678</Phone2>
    <Fax></Fax>
    <Email>moshe@cochavi.co.il</Email>
    <CreditCard>4580000000000000</CreditCard>
    <ExpDate>1013</ExpDate>
    <Comments></Comments>
    <OpenBalance>0</OpenBalance>
    <ForeginKey>39585.8553703704</ForeginKey>
    <BankAccountNumber>123456789</BankAccountNumber>
    <BankBranch>úi îâðã</BankBranch>
    <BanksID>24</BanksID>
    <PriceListNum>1</PriceListNum>
    <Charge>0</Charge>
    <ChargeCurrencyName>NIS</ChargeCurrencyName>
    <PaymentTerms>2</PaymentTerms>
    <PaymentDays>30</PaymentDays>
  </Customer>
  <Customer>
    <CustomersID>390</CustomersID>
    <BusinessName>îçñðé çùîî àò"î</BusinessName>
    <OsekMorshe>6464777</OsekMorshe>
```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

    <Address1>ú.ã. 728</Address1>
    <Address2></Address2>
    <City>úi àáéá</City>
    <PostalCode>72778</PostalCode>
    <ContactFirstName>éò÷á</ContactFirstName>
    <ContactLastName>îæøçé</ContactLastName>
    <Phone1>7272888</Phone1>
    <Phone2>828282</Phone2>
    <Fax>2828999</Fax>
    <Email>y@w.com</Email>
    <CreditCard>262727288</CreditCard>
    <ExpDate>0309</ExpDate>
    <Comments>iâiç iââçîâçâ âiçâiâçâiêâç âiçâiêâ çã</Comments>
    <OpenBalance>-788.23</OpenBalance>
    <ForeginKey>39586.8071412037</ForeginKey>
    <BankAccountNumber>12882/78</BankAccountNumber>
    <BankBranch>900</BankBranch>
    <BanksID>24</BanksID>
  </Customer>
</Caspit>

```

XML עבור יצירת חשבונית מס/קבלה

פרטי הלקוח נמסרים באמצעות ה-ID של הלקוח. כלומר, קודם כל יוצרים את הלקוח ואז שולחים את ה-ID ב-XML:

```

<Caspit>
<Document>
<CustomerID>370</CustomerID>
<TrxCodesID>13</TrxCodesID>
<DocDate>2008-05-27</DocDate>
<DocComments>הסחורה בבעלות אחים עמר עד לתשלום מלא</DocComments>
<DocDetails>עמר ציון מוכרן: ציון עמר</DocDetails>
<RebatePercent>0</RebatePercent>
<RebateAmount>0.00</RebateAmount>
<RoundTotal>False</RoundTotal>
<ChargeVAT>True</ChargeVAT>
<VAT>0.00</VAT>
<LineItems>
  <LineItem number="1">
    <ItemDescription>חפסת לייזר</ItemDescription>
    <ItemQuantity>1</ItemQuantity>
    <UnitPrice>86877.</UnitPrice>
    <Currency></Currency>
    <CurrencyRate>1.0</CurrencyRate>
  </LineItem>
</LineItems>
<Receipt>
  <Total>1003.43</Total>
  <Tax>100.00</Tax>
  <NetPmt>903.43</NetPmt>
  <ReceiptLineItems>
    <ReceiptLineItem number="1">
      <AccountNumber>111</AccountNumber>
      <Amount>903.43</Amount>
      <Bank>1</Bank>
      <Branch>222</Branch>
    </ReceiptLineItem>
  </ReceiptLineItems>
</Receipt>

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```

        <Check>3333</Check>
        <CheckDate>2009-01-19</CheckDate>
        <PaymentTypesID>2</PaymentTypesID>
    </ReceiptLineItem>
</ReceiptLineItems>
</Receipt>
</Document>
</Caspit>

```

XML עבור יצירת חשבונית זיכוי

```

<Caspit>
<Document type="Invoice">
<CustomerID>4</CustomerID>
<TrxCodesID>3</TrxCodesID>
<DocDate>2011-08-25</DocDate>
<DocComments></DocComments>
<DocDetails></DocDetails>
<RebatePercent>0</RebatePercent>
<RebateAmount>0.00</RebateAmount>
<RoundTotal>False</RoundTotal>
<ChargeVAT>True</ChargeVAT>
<InvoiceType>1</InvoiceType>
<VAT>0.00</VAT>
<Invoices>10000,10003,20004</Invoices> → מספרים של חשבונית מס וחס/קבלה
קשורים
<LineItems>
    <LineItem number="1">
        <ItemDescription>החזרות/זיכויים</ItemDescription>
        <ItemQuantity>1.3</ItemQuantity>
        <UnitPrice>1000</UnitPrice>
        <Currency></Currency>
        <CurrencyRate>1.0</CurrencyRate>
        <ChargeVAT>True</ChargeVAT>
    </LineItem>
    <LineItem number="2">
        <ItemDescription>זיכוי עבור הובלה</ItemDescription>
        <ItemQuantity>1</ItemQuantity>
        <UnitPrice>200</UnitPrice>
        <Currency></Currency>
        <CurrencyRate>1.0</CurrencyRate>
        <ChargeVAT>True</ChargeVAT>
    </LineItem>
</LineItems>
</Document>
</Caspit>

```

XML עבור סיווגי תנועה

דוגמה עבור רשימה של סיווגי התנועה כפי שנספית מחזירה אותם:

```

<Caspit>
    <TrxCodes>
        <TrxCode>
            <TrxCodesID>4</TrxCodesID> → פנימי של כספית ID
            <TrxCodeNumber>4</TrxCodeNumber> → מספר הסיווג
        </TrxCode>
    </TrxCodes>
</Caspit>

```

כל הזכויות שמורות © כספית תוכנה בע"מ

```
        <TrxCodeDescription> הכנסות/מכירות פטורות  
&quot;n</TrxCodeDescription> → תיאור  
        <PLCodesID>1</PLCodesID> → קוד רווח והפסד  
</TrxCode>  
<TrxCode>  
        <TrxCodesID>75</TrxCodesID>  
        <TrxCodeNumber>204</TrxCodeNumber>  
        <TrxCodeDescription> קניות ללא  
n"מ</TrxCodeDescription>  
        <PLCodesID>2</PLCodesID>  
</TrxCode>  
</TrxCodes>  
</Caspit>
```

דוגמאות קוד

הדוגמאות הן ב-VB 6. הערכים בחלק מדוגמאות נלקחים מטופס שהשתמש מילא. יש דוגמאות מקיפות יותר בפרויקט APIDEMO שמופק עם ה-API.

דוגמה ליצירת לקוח

```
'Create the customer
Set objCaspit = New CaspitAPI.Caspit
objCaspit.CaspitPath = g_strCaspitPath
objCaspit.AccessPath = g_strAccessPath
objCaspit.DebugMode = True
objCaspit.BusinessID = 3
```

```
iCustomersID = objCaspit.CreateCustomer(txtBusinessName, _
    txtOsekMorshe, _
    txtContactFirstName, txtContactLastName, _
    txtAddress1, txtAddress2, txtCity, txtPostalCode, _
    txtPhone1, txtPhone2, txtFax, txtEmail, _
    txtComments, CCur(txtOpenBalance), _
    txtAccountNumber, txtBankBranch, iBankID, _
    CLng(lstPriceListNum), CCur("0" & txtCharge), lstCurrency, _
    CStr(CDb1(Now)), iPaymentTermsID, iPaymentDays)
```

```
Set objCaspit = Nothing
```

דוגמה ליצירת חשבונית

```
Dim varLineItems(2) As CaspitAPI.LineItem
Dim objCaspit As CaspitAPI.Caspit
Dim I As Long

Set objCaspit = New CaspitAPI.Caspit

For I = 0 To 2
    Set varLineItems(I) = New CaspitAPI.LineItem
    varLineItems(I).ItemDescription = txtItemDescription(I)
    If Len(txtUnitPrice(I)) > 0 Then varLineItems(I).UnitPrice =
CCur(txtUnitPrice(I))
    If Len(txtItemQty(I)) > 0 Then varLineItems(I).ItemQuantity =
CDbl(txtItemQty(I))
    If Len(txtRate(I)) > 0 Then varLineItems(I).CurrencyRate =
CDbl(txtRate(I))
    varLineItems(I).ItemCurrency =
lstCurrency(I).List(lstCurrency(I).ListIndex)
Next

objCaspit.CaspitPath = g_strCaspitPath
objCaspit.DebugMode = g_blnDebug
objCaspit.BusinessID = 3

m iInvoiceNum = objCaspit.CreateInvoice(iCustomersID,
datInvoiceDate, _
    txtComments, txtRebatePercent, txtRebateNIS, chkRound.Value,
txtVAT, chkVAT.Value, _
```

```
txtDetails, _  
iTrxCodesID, _  
INVOICE TYPE STD, _  
varLineItems, _  
CDate(datDueDate))
```

```
Set objCaspit = Nothing
```

דוגמה להדפסת חשבונית באנגלית

```
Dim strInvoiceNum As String  
Dim objCaspit As caspitAPI.Caspit  
  
20 g_strCaspitPath = txtCaspitPath.Text  
30 g_blnDebug = chkDebug.Value = 1  
  
40 strInvoiceNum = InputBox("יחידות אצות")  
  
50 If LenB(strInvoiceNum) = 0 Then Exit Sub  
  
60 Set objCaspit = New caspitAPI.Caspit  
70 With objCaspit  
80 .CaspitPath = g_strCaspitPath  
90 .AccessPath = g_strAccessPath  
100 .BusinessID = g_iBusinessID  
110 .DebugMode = g_blnDebug  
120 End With  
130 Call objCaspit.PrintInvoice(CLng(strInvoiceNum),  
caspitAPI.Languages.English)  
140 Set objCaspit = Nothing
```

ה-API וה-Registry

ה-API משתמש ב- KEY הבא: HKEY_LOCAL_MACHINE\SOFTWARE\Caspit\API

הערכים שישנם ב-KEY הם:

- IsAPIDebug – האם ה-API מופעל במוד של DEBUG. השליטה היא דרך התכונה DEBUGMODE של ה-API.
- יתר הערכים (לדוגמא: IsAskForCopy) אינם בשימוש (3.2)
- כמו-כן ישנם מספר ערכים שקשורים ל-API ב-KEY: HKEY_LOCAL_MACHINE\SOFTWARE\Caspit
- APIVersion – גרסת ה-API שמותקנת במחשב. הרשומה נוצרת בזמן ההתקנה של כספית. כלומר, אם תתקין גרסה חדשה של ה-API בעזרת regsvr32 יישאר ברשומה מספר הגרסה הישן.
- File-ו-Path – כך ה-API מוצא את הכספית ברירת המחדל שהוא עובד מולה. אם הערכים האלה חסרים או ריקים ה-API לא יצליח לאתחל את עצמו ויציג הודעת שגיאה.

Web API

דרישות מוקדמות להפעלה מוצלחת של ה-API

להלן רשימה של דרישות מוקדמות בכדי שיהיה ניתן להפעיל בהצלחה את ה-API של כספית:

- ❖ התקנה תקינה של כספית.
כלומר כספית נפתחת באופן תקין וניתן להוסיף לה לקוחות, חשבונות וכו'.
חשוב לבדוק גם הדפסות. כלומר שניתן להדפיס באופן תקין מכספית.
אחת המשמעויות היא שיש התקנה של Access 2003 או מאוחר יותר על המחשב.
- ❖ התקנה של CaspitAPI32.dll – זהו המימוש של ה-API עבור גרסה 3.2 של כספית. מאחר שהקומפוננטה היא ActiveX יש להריץ עליה regsvr32. הקומפוננטה דרושה לצורך הקישור של מערכת ה-Web לכספית של המשתמש.
- ❖ התקנה של regobj.dll – קומפוננטה של מיקרוסופט לגישה ל-registry. יש להריץ עליה regsvr32.
- ❖ רישוי מתאים:

○ לצורך ההפעלה של Web API – רישיון הפעלה לכספית שכולל הרשאה לשימוש ב-Caspit.Web

מה הטכנולוגיה של ה-API?

ה-API ממומש כ-Web Service. בכדי להסביר בפשטות במה מדובר הנה דוגמה:

== יושלם בגרסה הבאה של התיעוד ==

כיצד מתחברים ל-API?

== יושלם בגרסה הבאה של התיעוד ==

יצירת חשבונות

== יושלם בגרסה הבאה של התיעוד ==

יצירת חש/קבלה

== יושלם בגרסה הבאה של התיעוד ==

שליפת לקוחות

== יושלם בגרסה הבאה של התיעוד ==

שליפת חשבונות

== יושלם בגרסה הבאה של התיעוד ==

שליפת חש/קבלה

== יושלם בגרסה הבאה של התיעוד ==

נספחים

נספח א – רשימת בנקים

הרשימה מעודכנת ל 26/3/2011. הרשימה משתנה לעיתים לפי עדכונים של בנק ישראל

Banks	
vcBankName	iBanksID
הבניה	1
צפון אמריקאי	2
יהב	4
בנק הדואר	9
בנק לאומי	10
בנק דיסקונט	11
בנק הפועלים	12
בנק איגוד	13
בנק אוצר החייל	14
מרכנתיל	17
בנק המזרחי	20
Citibank N.A	22
HSBC Bank plc	23
BNP Paribas Israel	25
יובנק	26
למסחר	30
בין לאומי	31
ערבי ישראלי	34
SBI State Bank of India	39
בנק מסד	46
פאי	52
בנק ירושליים	54
בנק דקסיה ישראל	68

נספח ב' – רשימת סוגי תשלום

Payment_types	
vcPaymentTypeName	iPaymentTypesID
מזומן	1
שיק	2
ויזה כאל	3
ויזה לאומי	4
ישראכרד	5

Payment_types	
vcPaymentTypeName	iPaymentTypesID
מסטרקרד	6
אמריקן אקספרס	7
כ"א אחר	8
העברה	9
הוראת קבע	10
תשלום מפקדון	11
אחר	12
שובר דואר	13
תשלום מקוון/ארנק אלקטרוני	14

נספח ג' – סטטוסים של מסמכים שנשלפים מכספית

משמעות	status text	iStatusID
המסמך פתוח לעריכה. מוצג באדום ברשימה בכספית.	פתוח	0
המסמך הודפס ואסור לשנות אותו. מוצג בירוק ברשימה בכספית.	נעול	2
המסמך בוטל על-ידי המשתמש. אסור לשנות אותו. מוצג בסגול ברשימה בכספית.	מבוטל	3

נספח ד' – סוגי מסמכים בכספית

Public Enum ObjectType

Invoice = 1

RECEIPT = 2

EXPENSES = 3 → הוצאה

HIYUV = 6 → חשבונית עסקה

INVOICE_KABALA = 7

INVOICE_ZIKUI = 9

REFUND = 13

ORDER_ = 14 → הזמנת עבודה

SHIPPING = 15 → תעודת משלוח

PROPOSAL = 16 → הצעת מחיר

End Enum

נספח ה' – רשימה של סוגי תנועה שמוגדרים בכספית

אלו סוגי התנועה שמוספים עם כספית. שם לב שהמשתמש יכול להגדיר סוגי תנועה בעצמו.

שם לב שה-ID של הסיווג (העמודה לא מוצגת כאן אבל קיימת בטבלה בכספית) של הסיווג "נזיל" במובן **שהוא משתנה מעסק לעסק** ויכול להשתנות בעסק נתון העקבות שיחזור של גיבוי.

מספר סיווג	תיאור
2	קבלה
3	הכנסות/מכירות

כל הזכויות שמורות © כספית תוכנה בע"מ

מספר סיווג	תיאור
4	הכנסות/מכירות פטורות ממע"מ
6	קבלות עבור תרומה
202	קניות של חומרי גלם
203	קניות של חומרי עזר
301	אחזקת משרד
302	אירוח במסעדות
303	ארנונה
304	ביגוד מקצועי
305	ביטוח של המשרד
306	ביטוחים לרכב וטסט
307	דואר
308	דלק לרכב
309	הוצאות מקצועיות (עו"ד, רו"ח וכו')
310	הוצאות משכורת ונלוות
311	הוצאות משרדיות
312	החזקת אופנוע (ביטוח, טסט)
313	החזקת אופנוע (תיקונים, דלק)
314	השת' בתערוכות מקצועיות
315	חניה וחניונים
316	חשמל
317	טלפון
318	טלפון נייד
319	כיבוד במשרד
320	לינה במלונות (בתפקיד)
321	מיסים ואגרות
322	מתנות ללקוחות ולספקים
323	נסיעות (מקצועיות!) לחו"ל
324	נסיעות במוניות (בתפקיד)
325	ספרות מיקצועית
326	עמלות
327	פירסום
328	פקס + חיבור לאינטרנט
329	קנסות
329	קורסים מקצועיים ולימודים
330	ריבית
331	שכירות/ליסינג רכב מסחרי
334	שכירת ליסינג רכב פרטי
335	שליחויות והובלות
336	תיקוני רכב
337	תרומות
338	תשלומים לקבלני משנה
339	הוצאות כלליות
900	קניית רכוש קבוע מוכר למע"מ

מספר סיווג	תיאור
901	קניית רכוש קבוע לא מוכר למע"מ
902	מכירת רכוש קבוע חייב מע"מ
903	מכירת רכוש קבוע - פטור ממע"מ
961	מלאי פתיחה
962	מלאי סגירה

האם המשתמש יכול לשנות את המספר של סיווג התנועה?

המספרים עם רקע אפור קבועים, מספרים עם רקע לבן – ניתן לשנות

טופס הגדרת סיווג תנועות						
מספר הסיווג קבועים ולא ניתנים לשינוי על-ידי המשתמש						
מס' סיווג	תיאור	סיווג רווח/הפסד	סיווג מע"מ	כמה מע"מ לחייב/לזכות	חלק שמוכר	מס' סיווג
2	קבלות	קבלות	לא משתתף בדו"ח מע"מ	אין	מלא	2
3	הכנסות/מכירות	הכנסות/מכירות	עסקאות חייבות במע"מ	מלא	מלא	3
4	הכנסות/מכירות פטורות ממע"מ	הכנסות/מכירות	עסקאות פטורות ממע"מ	אין	מלא	4
101	הכנסות סמויות	הכנסות/מכירות	עסקאות חייבות במע"מ	מלא	מלא	101
202	קצוות של חומרי גלם	תצרכת חומרים	מע"מ תשומות אחרות	מלא	מלא	202
203	קצוות של חומרי עזר	תצרכת חומרים	מע"מ תשומות אחרות	מלא	מלא	203
204	קצוות ללא מע"מ	תצרכת חומרים	לא משתתף בדו"ח מע"מ	אין	מלא	204
301	אזקות משרד	הוצאות מכירה, הנהלה	מע"מ תשומות אחרות	מלא	מלא	301
302	אזקות במסעדות	הוצאות מכירה, הנהלה	מע"מ תשומות אחרות	אין	מלא	302
303	אזקות	הוצאות מכירה, הנהלה	לא משתתף בדו"ח מע"מ	אין	מלא	303
304	בזק מקצועי	הוצאות מכירה, הנהלה	מע"מ תשומות אחרות	מלא	מלא	304
305	בטוח של המשרד	הוצאות מכירה, הנהלה	לא משתתף בדו"ח מע"מ	אין	מלא	305
306	בטוחים לרכב וטסט	הוצאות מכירה, הנהלה	לא משתתף בדו"ח מע"מ	אין	מלא	306
307	דאר	הוצאות מכירה, הנהלה	מע"מ תשומות אחרות	מלא	מלא	307
308	דלק לרכב	הוצאות מכירה, הנהלה	מע"מ תשומות אחרות	2/3	מלא	308