**מסמך ייזום**

**מסמך יזום**

**תיאור הארגון :** הארגון הוא ספרייה בבעלות פרטית שמתפקדת כספרית עיון ו השאלה . הספרייה מספקת לבאים בשעריה שרותי מידע ללימוד ו העשרה , שרותי קריאה לשעות הפנאי לילדים , נוער ו מבוגרים בשלוש שפות עיקריות ערבית , עברית ו אנגלית .

**מטרות הארגון :**

מטרות העל של הארגון היא לספק שירותי קריאה עיון ו השאלה בצורה הנוחה ו הטובה ביותר ו כמובן להבי רווח מרבי לבעליה ו עובדיה . מטרות המשנה של הארגון 1- לעודד מודעות , סקרנות ו הערכה למורשת התרבותית ו צריכת ידע בתחומי תרבות , אמנויות ו מדעים . 2- עידוד קריאה בקהילה ו פיתוח מודעות כלפי חשיבות הקיאה ו יתרונותיה . 3-לשרת את הקהילה על בסיס עקרונות הדמוקרטיה בתחומי חופש הביטוי , הפלורליזם ו שוויון הזדמנויות.

**היקף הפעילות:**

ישנם שלושה סניפים לספריה ,הראשון ממוקם בתל אביב מיקומו חיוני ו ער ,הסניף פתוח משעה 7:00 עד שעה 21:00 כל ימי השבוע למעט החגים ו ימי שבת ,הסניף השני בבר שבע ו השלישי בנתניה ,הסניפים פתוחים משעה 5:00 עד 19:00 כל ימי השבוע למעט החגים . מושאלים ביום כ70 ספרים בכל סניף ו מוחזרים כ40 ספרים באים ביום כ100 איש מהמם תלמידי תיכון ילדים בגילאים 6-10 , מבוגרים ו רובם סטודנטים , שבאים ליהנות בקריאת ספרים מעניינים ו מאתגרים ו לצורך למידה . הספרייה מספקת לבאים בשעריה מענה מקצועי בהתאם לצרכים ו לתחומי העניין כמו כן דואגת לסביבה נוחה ו שקטה לאנשים ו בה יש מחשיבים ו טאבלטים חדשים שמשרתים את האנשים בעט הצורך ו בנוסף לכך יש בית קפה קטן ליד הספרייה ששייך אליה לצורך מנוחה.

**מבנה הארגון:**

בכל סניף מועסקים ביום 12 , במשמרת הראשונה מ 7:00 עד 14:00 יש 2 סדרניות ו 3 ספרניות ,שתיים מהן למדור הילדים ו אחת למבוגרים ו עובד אחד שמנהל את ההשאלות ו ההזמנות של הספרייה באמצעות מערכת ממוחשבת ו במשמרת השנייה מ- 14:00 עד 21:00 מתחלפים העובדים באותו אופן . לכל שלושת הסניפים ישנו מנהל אחד שקובע את החברות שמספקות ספרים ו את כמות הספרים שמזמינים לספריה .

**הנהלה**

**מנהל כל הסניפים**

**מנהל סניף**

**מהות המשימה ו הרקע שלה :**

במערכת המידע הידנית של הארגון קיימות מספר בעיות : 1- השאלות : במערכת המידע הידנית השאלות נרשמות על נייר בקלסרים ו לפעמים על ניירת ו זה עלול לגרום לאובדן מידע שגורם לאיבוד רכוש הספרייה אם לא נדע אם הספרים הוחזרואו לא 2- אבדן מידע: הספרייה מפסידה הרבה כסף כתוצאה מאיבוד רכוש עקב אובדנו של המידע על ההשאלות ו ההחזרות . 3- הזמנות: במערכת המידע הידנית ההזמנות נרשמות על יומן ו לפעמים על ניירת וזה גורם לאובדן מידע הקשור להזמנות ו זה גורם לתלונות מן הלקוחות על אי קבלת הזמנות . 4-סיכומים:בהרבה מקרים צריך לעשות דוחות לסיכום ו זה לא אפשרי במערכת המידע הידנית כי המידע מפוזר בהרבה ניירות ו גם יש אובדן ניירת ו זה מסיבת שיטת האחסונים בקלסרים של ניירת . 5-חברות מספקות :במערכת המידע הידנית פרטי החברות שמספקות את הספרים לספריה נרשמות על נייר בקלסרים או ניירת ו זה עלול לגרום לאובדן מידע לגביהם ו דבר זה עלול לשבש את פעולת ההזמנה של הספרים .

**מטרות יעדים ו מדדים**

**מטרה 1-**סיוע בניהול ההשאלות ו ההחזרות . יעד – מעקב ממוחשב להשאלות ו ההחזרות . מדד – המערכת תציג דוחות על ההשאלות ו ההחזרות ו תציג התרעות אם לקוח לא החזיר ספר על לתאריך האחרון .

**מטרה 2 –** סיוע בניהול ההזמנות של הלקוחות ו הזמנות הספרייה . יעד – מעקב ממוחשב על ההזמנות של הלקוחות ו הזמנות ספרים לספרייה . מדד- המערכת תציג דוחות על ההזמנות ו כאשר מגיעה הזמנה המערכת מציגה הודעה כדי להודיע ללקוח או כדי לקבל הספרים שהגיעו לספריה.

**מטרה 3-** סיוע בשמירה על המידע אודות חברות הסיפוק . יעד- שמירת נתוני החברות במחשב ו רק המנהל יכול לעדכן שינוים בפרטים.  מדד- המערכת תשמור את נתוני החברות במאגרי מידע ו לא תאפשר ביצוע שינויים בפרטים , רק על ידי המנהל הראשי  **.**

**תיחום**

המערכת תטפל ב: -ההשאלות ו ההחזרות שמבציע הלקוח -איחורים בהחזרת ספרים -הזמנות ספרים חדשים לספריה -הזמנת ספרים ללקוחות ממאגר הספרים שבספריה -פתיחת כרטיס לקוח לספרייה - מאגר הספרים - חברות שמספקות ספרים לספרייה -דוחות על ההשאלות , החזרות , הזמנות , ספקים , ספרים

המערכת לא תטפל ב : -תשלום דמי ההרשמה -תשלום דמי הזמנת ספרים חדשים שלא נמצאים בספריה

**אילוצים**

1-אילוץ כוח אדם : - המערכת תפתח רק על ידי מנהל הסניף או על ידי המנהל הראשי - פיתוח המערכת יעשה על ידי מהנדס התוכנה שיכול להיעזר בבעל העסק - המפתחים יכולים להתייעץ עם גורמי חוץ ,אבל לא להעסיקם על בסיס קבוע

2-אילוץ תקציב: - הסכום המוקצה לרכישת התוכנה לא יעלה על 8000 שקל - הסכום המוקצה לחומרה ו לפיתוח התוכנה לא יעלה על 4000 שקל - תקציב התחזוקה השוטפת של המערכת לאחר הפעלתה לא יעלה על 1000 שקל לשנה

**הנחות**

1- מאגר הספרים יכול לגדול 2- מספר הלקוחות יגדל

**עץ**

**תהליכים**

**עץ תהליכים**

**P5.5**

**הצגת ספק**

**חיפוש ספק**

**P5.4**

**מחיקת ספק**

**P5.3**

**עדכון פרטי ספק**

**P5.2**

**הוספת ספק**

**P5.1**

**טיפול בספקים**

**הצגת הזמנה**

**P4.5**

**חיפוש הזמנה**

**P4.4**

**P4.3**

**ביטול ההזמנה מהספרייה**

**עדכון פרטי ההזמנה מהספריהה**

**P4.2**

**ביצוע הזמנה**

**P4.1**

**טיפול בהזמנות**

**P4**

**P5**

**בדיקת החזרות**

**P3.5 1**

**ביצוע השאלה**

**הצגת השאלה**

**P3.4 1**

**חיפוש השאלה**

**P3.3 1**

**עדכון פרטי השאלה**

**P3.2**

**P3.1**

**הצגת ספר**

**P2.4**

**חיפוש ספר**

**P2.3**

**עדכון פרטי הספר**

**P2.2**

**הוספת ספר לאוסף**

**P2.1**

**טיפול בספרים**

**P2**

**טיפול בהשאלות**

**P3**

**P1.6**

**הצגת לקוחות**

**חיפוש לקוח**

**P1.5**

**בדיקת פרטי לקוח**

**P1.4**

**סגירת מנוי**

**P1.3**

**עדכון פרטי לקוח**

**P1.2**

**הוספת לקוח**

**P1.1**

**P1**

**טיפול בלקוחות**

**P**

**ניהול ספרייה**

**DFD**

**0**

**P1**

מנהל הסניף מקבל את פרטי הלקוח ו מוסיף אותו לרשימת הלקוחות , או מעדכן את הפרטים במידה ו הלקוח קיים , ו מפיק דוחות על מצבם של הלקוחות .

מצב לקוח

פרטי לקוח

בקשת מידע על הלקוח

**מנהל סניף**

**E2**

פרטי לקוח פרטי השאלה פרטי הזמנה

**מאגר לקוחות**

**E1**

**לקוח**

**טיפול בלקוח**

**P1**

* - תהליך ידני איטי , מסורבל דורש הרבה זמן לאיתור לקוח - אין ביטול לקוח בעייתי אחרי כמה זמן - חזרה על אותם לקוחות הרבה פעמים באותו קלסר

**2P**

מנהל הסניף נכנס לרשימת הספרים מחפש ספר , מוסיף ספר למאגר הספרים ו מעדכן את פרטי הספר אם יש צורך .

מצב ספרים

פרטי ספרים

פרטי ספרים

מצב ספרים

**E1**

**מאגר ספרים**

**מנהל סניף**

**P2**

**טיפול בספרים**

-תהליך ידני ו איטי - עדכון הספרים לוקח זמן רה

**P3**

מנהל הסניף מקבל מהלקוח את ת.ז שלו מחפש ו בוחר את הספר שהלקוח רוצה מרשימת הספרים הזמינים .

בקשת מידע על ההשאלה

פרטי השאלה

פרטי השאלה

פרטי השאלה

**E1**

**E2**

**לקוח**

**מנהל סניף**

**מאגר ההשאלות**

**P3**

**טיפול בהשאלות**

- תהליך ידני ו איטי - איבוד נתונים עם איבוד הניירת (אותו דבר בהזמנות)

**P4**

מנהל הסניף מקבל מהלקוח את ת.ז שלו מחפש ו בוחר את הספר שהלקוח רוצה מרשימת הספרים ו אם הספר לא זמין באותו זמן אז מזמין אותו ו במידה ו הספר הזה לא קיים בספריה אז יש אפשרות הזמנת הספר מהספקים של הספרייה.

**E2**

**מנהל סניף**

**E1**

**לקוח**

פרטי הזמנה

בקשת מידע על ההזמנה

**P4**

**טיפול בהזמנות**

פרטי הזמנה

פרטי הזמנה

**טיפול בהזמנות של ספרים חדשים לספרייה**

**מנהל ראשי**

**E1**

פרטי הזמנה

**מאגר ההזמנות**

**P5**

המנהל הראשי מקבל את פרטי החברה שמספקת ו מוסיף אותה לרשימת הספקים, או מעדכן את הפרטים במידה ו הספק קיים , ו מפיק דוחות על מצבם של הספקים.

מצב ספק

פרטי ספק

**מאגר ספקים**

בקשת מידע על הספק

פרטי ספק

**ספק**

**מנהל ראשי**

**E2**

**P5**

**טיפול בספקים**

-תהליך ידני ו איטי - עדכון הספרים לוקח זמן רה

**ריכוזבעיות**

**ריכוז בעיות**

שם הבעיה : מהלך איטי ו לוקח זמן . תיאור : יש מהלכים שלוקחים הרבה זמן עד לביצועם ו גם איטיים ו זה בגלל שמהלכים אלה ידניים . רמה: 10\9 פתרון : לעבור למערכת ממוחשבת .

שם בעיה : קושי במציאת פרטי השאלות ו הזמנות . תיאור : כאשר מנהל הסניף מבקש פרטי הזמנה או השאלה מסוימת ,נתקלים בבעיה במציאת פרטי ההזמנה או ההשאלה המבוקשת וזה גורם לבלבול ו בעיות בערכת. רמה :10\6 פתרון : להכין לכל השאלה או הזמנה תיקיה ו לשמור בה פרטי ההזמנה או ההשאלה .

שם בעיה : הפרטים נכתבים בכתב יד לא ברור . תיאור : כאשר רוצים לקרוא פרטים מסוימים נתקלים בבעיה בקריאה ולא מבינים מה כתוב בגלל שהפרטים נכתבו בכתב יד לא ברור . רמה : 10\8 פתרון : להכניס את הנתונים למחשב ו להדפיס את מה שהוכנס אחר כך .

שם בעיה : קושי בהכנת דיווח להשאלות ו הזמנות . תיאור: כאשר המנהל הראשי מבקש דיווח להשאלות או להחזרות נתקלים בבעיה במציאת פרטי ההזמנה או ההשאלה המבוקשת ו לפעמים לא מוצאים בכלל . רמה : 10\6 פתרון : להכין עותק לכל דיווח .

**תיאור מאגרים קיים**

תיאור מאגרים קיים

שם מאגר : לקוחות (כרטיס קורא) סוג מאגר : ידני קוד מאגר : DS1 היקף : כ 1300 רשומות (לפי מספר הלקוחות בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז , שם , שם משפחה , טלפון , תאריך פתיחת מנוי , תאריך הסתיימות המנוי ,אם הלקוח חייב כסף לספרייה , אם המנוי בתוקף . בעיות : - קושי במציאת פרטי לקוח - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ...

|  |  |  |
| --- | --- | --- |
| **שם שדה** | **טיפוס** | **אורך שדה** |
| ת.ז | נומרי | 9 |
| שם | מחרוזת | 15 |
| שם משפחה | מחרוזת | 15 |
| טלפון | נומרי | 10 |
| תאריך פתיחת המנוי | מחרוזת | 10 |
| תאריך הסתיימות המנוי | מחרוזת | 10 |
| חייב | נומרי | 3 |
| מנוי בתוקף | מחרוזת  סה"כ תווים : 74  סה"כ רשומות : 1300  סה"כ נפח איחסון : 96,200 | 2 |

שם מאגר : ספרים סוג מאגר : ידני קוד מאגר : DS2 היקף : כ 2300 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : קוד , שם , שם המחבר, תאריך ההוצאה, שפה , כמות ספרים בספרייה , סוג , גיל מתאים , אם הספר מסומן לא להשאלה , חברה מספקת ו עותקים בעיות : - קושי במציאת פרטי הספר - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ...

|  |  |  |
| --- | --- | --- |
| **שם שדה** | **טיפוס** | **אורך שדה** |
| קוד | נומרי | 6 |
| שם | מחרוזת | 15 |
| שם המחבר | מחרוזת | 25 |
| שנת ההוצאה | נומרי | 4 |
| שפה | מחרוזת | 6 |
| כמות ספרים זמינים | נומרי | 2 |
| סוג | מחרוזת | 15 |
| גיל מתאים | מחרוזת | 25 |
| מסומן | מחרוזת | 2 |
| חברה | מחרוזת | 15 |
| עותקים בספרייה | נומרי  סה"כ תווים : 117  סה"כ רשומות : 2300  סה"כ נפח איחסון : 269,100 | 2 |

שם מאגר :השאלות סוג מאגר : ידני קוד מאגר : DS3 היקף : כ 1500 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז קורא , שם ספר , תאריך ההשאלה בעיות : - קושי במציאת פרטי השאלות - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ... –קושי במעכב אחרי ההחזרות כאשר מגיעים לתאריך האחרון ו חודש

|  |  |  |
| --- | --- | --- |
| שם שדה | טיפוס | אורך שדה |
| ת.ז קורא | נומרי | 9 |
| שם ספר | מחרוזת | 30 |
| תאריך ההשאלה | מחרוזת  סה"כ תווים : 49  סה"כ רשומות : 1500  סה"כ נפח איחסון : 73,500 | 10 |

שם מאגר :הזמנות ספר ממאגר הספרים בספריה סוג מאגר : ידני קוד מאגר : DS4 היקף : כ 300 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז קורא , שם ספר , תאריך ההזמה ו תאריך קבלת ההודעה שההזמנה מוכנה בעיות : - קושי במציאת פרטי ההזמנות - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ... – קושי במעכב אחרי ההזמנה ו מתי תהייה מוכנה כלומר הספר יהיה קיים זמין

|  |  |  |
| --- | --- | --- |
| שם שדה | טיפוס | אורך שדה |
| ת.ז קורא | נומרי | 9 |
| שם ספר | מחרוזת | 30 |
| תאריך ההזמנה | מחרוזת | 10 |
| תאריך קבלת הודעה | מחרוזת  סה"כ תווים : 59  סה"כ רשומות : 300  סה"כ נפח איחסון : 17,700 | 10 |

שם מאגר :הזמנות ספר חדשים לספרייה סוג מאגר : ידני קוד מאגר : DS5 היקף : כ 350רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : שם ספר , שם החברה שמספקת את הספר , תאריך ההזמנה , כמות ספרים שהוזמנו בעיות : - קושי במציאת פרטי ההזמנות - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ... – קושי במעכב אחרי ההזמנה ו מתי תגיע לספריה - קושי בבדיקה אם הספרייה עובדת עם החברה של הספר (כלומר החברה נמצאת במאגר הספקים של הספרייה)

|  |  |  |
| --- | --- | --- |
| **שדה** | **טיפוס** | **אורך שדה** |
| שם הספר | מחרוזת | 25 |
| שם החברה של הספר | מחרוזת | 25 |
| תאריך ההזמנה | מחרוזת | 10 |
| כמות ההזמנה | נומרי  סה"כ תווים : 62  סה"כ רשומות : 350  סה"כ נפח איחסון : 21,700 | 2 |

שם מאגר :החברות מספקות סוג מאגר : ידני קוד מאגר : DS6 היקף : כ 50רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : שם החברה , שם מנהל החברה , טלפון חברה , טלפון מנהל , מייל בעיות : - קושי במציאת פרטי הספקים - הפרטים נכתבים בכתב יד לא ברור - הדפים בקלסר מפוזרים - המהלך הזה לוקח זמן ו איטי ...

|  |  |  |
| --- | --- | --- |
| **שדה** | **טיפוס** | **אורך שדה** |
| שה חברה | מחרוזת | 25 |
| שם מנהל החברה | מחרוזת | 25 |
| טלפון החברה | נומרי | 10 |
| טלפון המנהל | נומרי | 10 |
| מייל | מחרוזת  סה"כ תווים : 110  סה"כ רשומות : 50  סה"כ נפח איחסון : 5,500 | 40 |

**יעדי המערכת העתידית**

יעדי המערכת העתידית

**יעד מס' 1:**

שם יעד: קליטה / עדכון לכל המאגרים תיאור: המערכת תסייע בניהול השוטף של מאגרי המערכת (קליטה ו עדכון). בעיות קשורות: קשה לאתר ו לשלוף מידע (כי המידע נמצא בקלסרים). אילוצים: אין במצב הקיים : 1- מאגרי מידע מאוחסנים בקלסרים 2- מידע חסר ו לא שלם 3- כתב יד אינו קריא

**יעד מס' 2:**

שם יעד: עדכון מאגר הלקוחות מהיר . תיאור: הכנת תהליך ממוחשב "עדכון פרטי לקוח " שיטפל בעדכון פרטי הלקוחות במאגר הלקוחות בעיות קשורות: עדכון מאגר הלקוחות איטי . אילוצים: אין במצב הקיים : 1- עדכון פרטי הלקוחות נעשה באופן ידני 2- המידע מתויק בקלסרים ו יכולת אובדן המידע גבוהה

**יעד מס' 3:**

שם יעד: שמירת תאריך ההשאלות ו ההזמנות במדויק. תיאור: הכנת תהליך שישמור את בזמן המדויק של ההשאלות ו ההזמנות בעיות קשורות: התאריכים הנרשמים היו שגויים לפעמים. אילוצים: אין במצב הקיים : 1-תאריך ההזמנה או ההשאלה היה נעשה בכתב יד שלפעמים לא קריא .

**יעד מס'4:**

שם יעד: מניעת חזרה על אותן רשומות במאגר . תיאור: למנוע חזרה על אותן רשומות , על אותן נתונים שבמאגרים . בעיות קשורות: רשומות חוזרות על עצמן לפעמים . אילוצים: אין במצב הקיים : 1-במאגראים יש חזרות על רשומות מסוימות.

**יעד מס'5:**

שם יעד: הכנת דוחות במהירות . תיאור: לערוך תהליך ממוחשב להפקת דווח. בעיות קשורות: קשה מאוד ו לקח זמן להכין דוחות מכיוון שהמידע מפוזר. אילוצים: אין במצב הקיים : 1- הכנת דוחות נעשה באופן ידני .

**עץ תהליכים עתידי**

**עץ תהליכים עתידי**

**P5.5**

**הצגת ספק**

**חיפוש ספק**

**P5.4**

**מחיקת ספק**

**P5.3**

**עדכון פרטי ספק**

**P5.2**

**הוספת ספק**

**P5.1**

**טיפול בספקים**

**הצגת הזמנה**

**P4.5**

**חיפוש הזמנה**

**P4.4**

**P4.3**

**ביטול ההזמנה מהספרייה**

**עדכון פרטי ההזמנה מהספריהה**

**P4.2**

**ביצוע הזמנה**

**P4.1**

**טיפול בהזמנות**

**P4**

**P5**

**בדיקת החזרות**

**P3.5 1**

**ביצוע השאלה**

**הצגת השאלה**

**P3.4 1**

**חיפוש השאלה**

**P3.3 1**

**עדכון פרטי השאלה**

**P3.2**

**P3.1**

**הצגת ספר**

**P2.4**

**חיפוש ספר**

**P2.3**

**עדכון פרטי הספר**

**P2.2**

**הוספת ספר לאוסף**

**P2.1**

**טיפול בספרים**

**P2**

**טיפול בהשאלות**

**P3**

**P1.6**

**הצגת לקוחות**

**חיפוש לקוח**

**P1.5**

**בדיקת פרטי לקוח**

**P1.4**

**סגירת מנוי**

**P1.3**

**עדכון פרטי לקוח**

**P1.2**

**הוספת לקוח**

**P1.1**

**P1**

**טיפול בלקוחות**

**P**

**ניהול ספרייה**

**תיאור מאגרים**

תיאור מאגרים קיים

שם מאגר : לקוחות (כרטיס קורא) סוג מאגר : ממוחשב קוד מאגר : DS1 היקף : כ 1300 רשומות (לפי מספר הלקוחות בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז , שם , שם משפחה , טלפון , תאריך פתיחת מנוי , תאריך הסתיימות המנוי ,אם הלקוח חייב כסף לספרייה , אם המנוי בתוקף .

|  |  |  |
| --- | --- | --- |
| **שם שדה** | **טיפוס** | **אורך שדה** |
| ת.ז | נומרי | 9 |
| שם | מחרוזת | 15 |
| שם משפחה | מחרוזת | 15 |
| טלפון | נומרי | 10 |
| תאריך פתיחת המנוי | מחרוזת | 10 |
| תאריך הסתיימות המנוי | מחרוזת | 10 |
| חייב | נומרי | 3 |
| מנוי בתוקף | מחרוזת  סה"כ תווים : 74  סה"כ רשומות : 1300  סה"כ נפח איחסון : 96,200 | 2 |

שם מאגר : ספרים סוג מאגר : ממוחשב קוד מאגר : DS2 היקף : כ 2300 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : קוד , שם , שם המחבר, תאריך ההוצאה, שפה , כמות ספרים בספרייה , סוג , גיל מתאים , אם הספר מסומן לא להשאלה , חברה מספקת ו עותקים

|  |  |  |
| --- | --- | --- |
| **שם שדה** | **טיפוס** | **אורך שדה** |
| קוד | נומרי | 6 |
| שם | מחרוזת | 15 |
| שם המחבר | מחרוזת | 25 |
| שנת ההוצאה | נומרי | 4 |
| שפה | מחרוזת | 6 |
| כמות ספרים זמינים | נומרי | 2 |
| סוג | מחרוזת | 15 |
| גיל מתאים | מחרוזת | 25 |
| מסומן | מחרוזת | 2 |
| חברה | מחרוזת | 15 |
| עותקים בספרייה | נומרי  סה"כ תווים : 117  סה"כ רשומות : 2300  סה"כ נפח איחסון : 269,100 | 2 |

שם מאגר :השאלות סוג מאגר : ממוחשב קוד מאגר : DS3 היקף : כ 1500 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז קורא , שם ספר , תאריך ההשאלה

|  |  |  |
| --- | --- | --- |
| שם שדה | טיפוס | אורך שדה |
| ת.ז קורא | נומרי | 9 |
| שם ספר | מחרוזת | 30 |
| תאריך ההשאלה | מחרוזת  סה"כ תווים : 49  סה"כ רשומות : 1500  סה"כ נפח איחסון : 73,500 | 10 |

שם מאגר :הזמנות ספר ממאגר הספרים בספריה סוג מאגר : ממוחשב קוד מאגר : DS4 היקף : כ 300 רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : ת.ז קורא , שם ספר , תאריך ההזמה ו תאריך קבלת ההודעה שההזמנה מוכנה

|  |  |  |
| --- | --- | --- |
| שם שדה | טיפוס | אורך שדה |
| ת.ז קורא | נומרי | 9 |
| שם ספר | מחרוזת | 30 |
| תאריך ההזמנה | מחרוזת | 10 |
| תאריך קבלת הודעה | מחרוזת  סה"כ תווים : 59  סה"כ רשומות : 300  סה"כ נפח איחסון : 17,700 | 10 |

שם מאגר :הזמנות ספר חדשים לספרייה סוג מאגר :ממוחשב קוד מאגר : DS5 היקף : כ 350רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : שם ספר , שם החברה שמספקת את הספר , תאריך ההזמנה , כמות ספרים שהוזמנו

|  |  |  |
| --- | --- | --- |
| **שדה** | **טיפוס** | **אורך שדה** |
| שם הספר | מחרוזת | 25 |
| שם החברה של הספר | מחרוזת | 25 |
| תאריך ההזמנה | מחרוזת | 10 |
| כמות ההזמנה | נומרי  סה"כ תווים : 62  סה"כ רשומות : 350  סה"כ נפח איחסון : 21,700 | 2 |

שם מאגר :החברות מספקות סוג מאגר :ממוחשב קוד מאגר : DS6 היקף : כ 50רשומות (לפי מספר הספרים בשנה) אמצע אחסון : קלסר שדות במאגר : שם החברה , שם מנהל החברה , טלפון חברה , טלפון מנהל , מייל

|  |  |  |
| --- | --- | --- |
| **שדה** | **טיפוס** | **אורך שדה** |
| שה חברה | מחרוזת | 25 |
| שם מנהל החברה | מחרוזת | 25 |
| טלפון החברה | נומרי | 10 |
| טלפון המנהל | נומרי | 10 |
| מייל | מחרוזת  סה"כ תווים : 110  סה"כ רשומות : 50  סה"כ נפח איחסון : 5,500 | 40 |

**DFD**

**0**

**P1**

מנהל הסניף מקבל את פרטי הלקוח ו מוסיף אותו לרשימת הלקוחות , או מעדכן את הפרטים במידה ו הלקוח קיים , ו מפיק דוחות על מצבם של הלקוחות .

מצב לקוח

פרטי לקוח

בקשת מידע על הלקוח

**מנהל סניף**

**E2**

פרטי לקוח פרטי השאלה פרטי הזמנה

**מאגר לקוחות**

**E1**

**לקוח**

**טיפול בלקוח**

**P1**

**2P**

מנהל הסניף נכנס לרשימת הספרים מחפש ספר , מוסיף ספר למאגר הספרים ו מעדכן את פרטי הספר אם יש צורך .

מצב ספרים

פרטי ספרים

פרטי ספרים

מצב ספרים

**E1**

**מאגר ספרים**

**מנהל סניף**

**P2**

**טיפול בספרים**

**P3**

מנהל הסניף מקבל מהלקוח את ת.ז שלו מחפש ו בוחר את הספר שהלקוח רוצה מרשימת הספרים הזמינים .

בקשת מידע על ההשאלה

פרטי השאלה

פרטי השאלה

פרטי השאלה

**E1**

**E2**

**לקוח**

**מנהל סניף**

**מאגר ההשאלות**

**P3**

**טיפול בהשאלות**

**P4**

מנהל הסניף מקבל מהלקוח את ת.ז שלו מחפש ו בוחר את הספר שהלקוח רוצה מרשימת הספרים ו אם הספר לא זמין באותו זמן אז מזמין אותו ו במידה ו הספר הזה לא קיים בספריה אז יש אפשרות הזמנת הספר מהספקים של הספרייה.

**E2**

**מנהל סניף**

**E1**

**לקוח**

פרטי הזמנה

בקשת מידע על ההזמנה

**P4**

**טיפול בהזמנות**

פרטי הזמנה

**E1**

**מאגר הזמנות**

**מנהל סניף**

פרטי הזמנה

**טיפול בהזמנות של ספרים חדשים לספרייה**

**מנהל ראשי**

**E1**

פרטי הזמנה

**מאגר ההזמנות**

**P5**

המנהל הראשי מקבל את פרטי החברה שמספקת ו מוסיף אותה לרשימת הספקים, או מעדכן את הפרטים במידה ו הספק קיים , ו מפיק דוחות על מצבם של הספקים.

מצב ספק

פרטי ספק

**מאגר ספקים**

בקשת מידע על הספק

פרטי ספק

**ספק**

**מנהל ראשי**

**E2**

**P5**

**טיפול בספקים**

**DFD 1**

**פרטי לקוח חדש**

**פרטי לקוח**

**מאגר לקוחות**

**P1.4**

**אישור הוספת לקוח**

**P1.3**

**שמירה/ עדכון פרטי לקוח**

**P1.2**

**בדיקת פרטי לקוח**

**P1.1**

**קליטת פרטי לקוח**

**E1**

**לקוח**

**פרטי ספר חדש**

**פרטי ספר**

**מאגר ספרים**

**P2.4**

**אישור הוספת ספר**

**P2.3**

**שמירה/ עדכון פרטי ספר**

**P2.2**

**בדיקת פרטי ספר**

**P2.1**

**קליטת פרטי ספר**

**E2**

**מנהל סניף**

**פרטי השאלה**

**מאגר ההשאלות**

**פרטי השאלה**

**P3.3**

**אישור פרטי השאלה**

**P3.2**

**בדיקת פרטי השאלה**

**P3.1**

**קליטת פרטי השאלה**

**E2**

**מנהל סניף**

**פרטי ההזמנות**

**פרטי ההזמנה**

**מאגר ההזמנות**

**P4.3**

**אישור פרטי ההזמנה**

**P4.2**

**בדיקת פרטי ההזמנה**

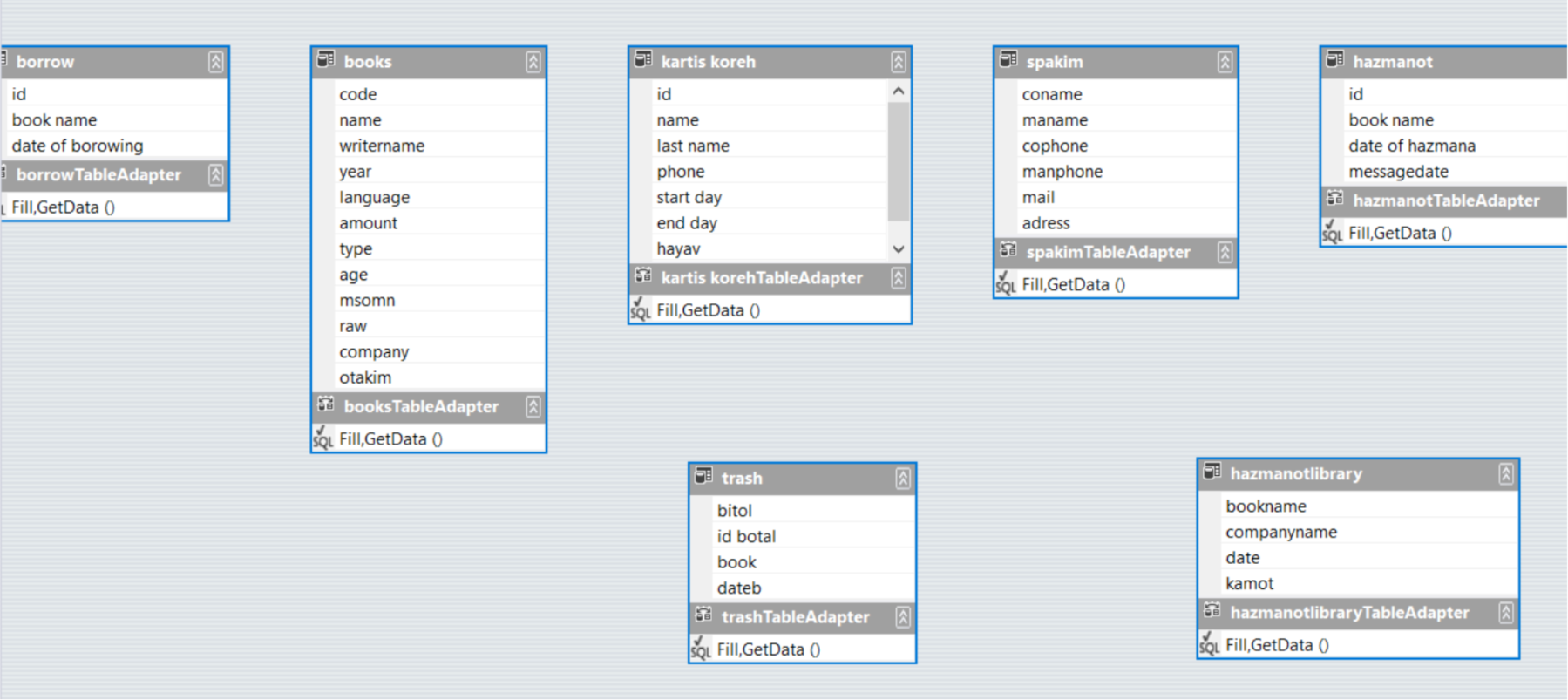
**P4.1**

**קליטת פרטי ההזמנה**

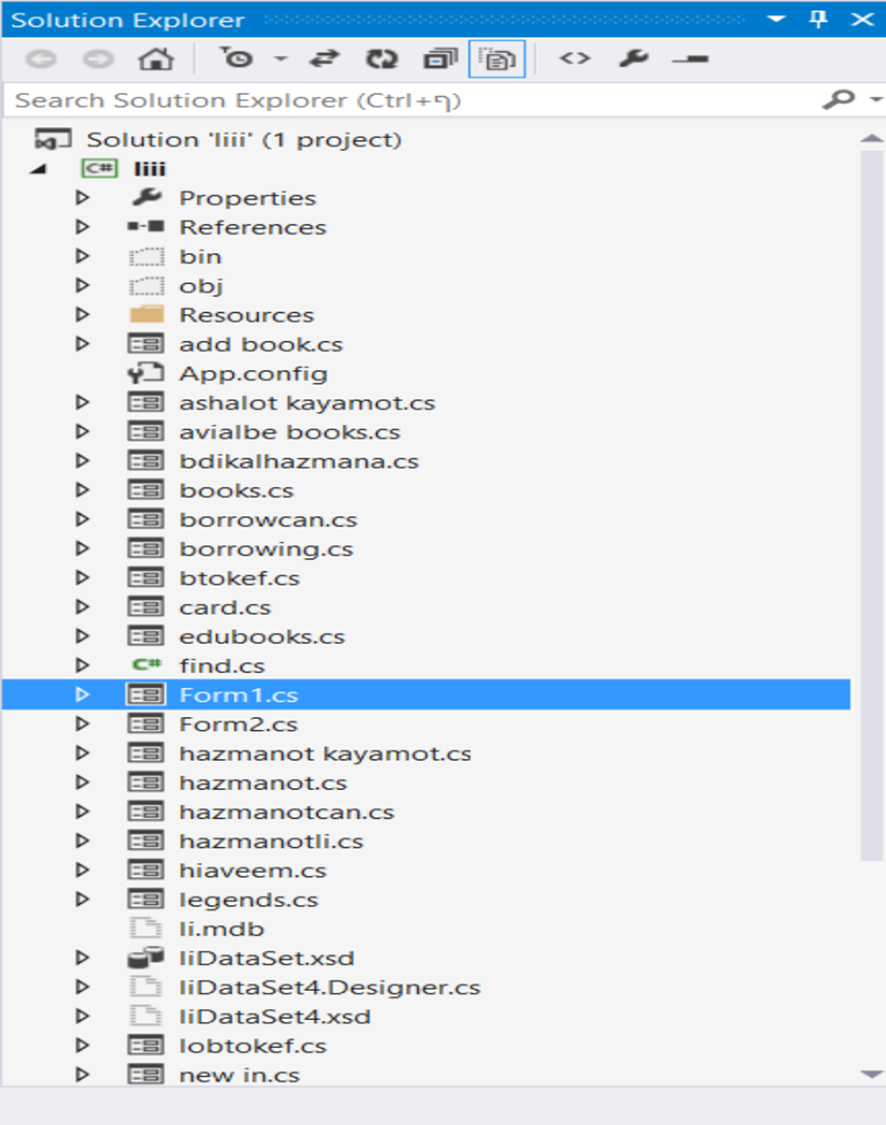
**E2**

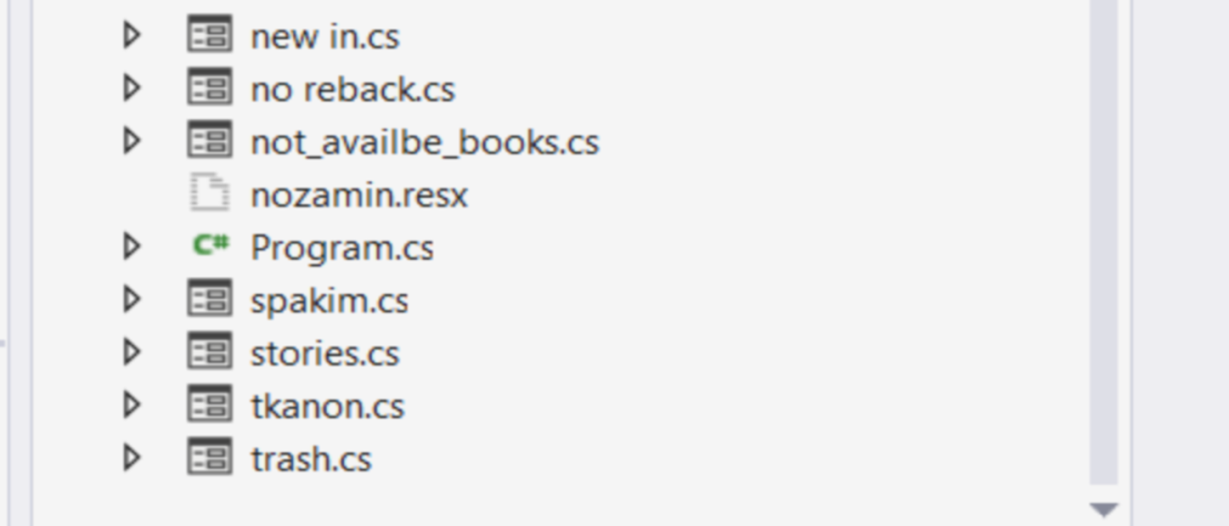
**מנהל סניף**

**טבלאות**



**טפסים**

**טפסים**

****

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class add\_book : Form

{

public add\_book()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.spakimDataTable kr;

liDataSet.booksDataTable kr3;

private void button1\_Click(object sender, EventArgs e)

{

if (maskedTextBox1.Text.Length != 6 || maskedTextBox1.Text.IndexOf(" ") > -1)

{

MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו ");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

if (textBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox2.Focus();

}

else

{

if (textBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox3.Focus();

}

else

{

if (textBox4.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox4.Focus();

}

else

{

int num = Convert.ToInt32(textBox4.Text);

if (num > 2018 || num<1800)

{

MessageBox.Show("נתון שגוי ");

textBox4.Focus();

textBox4.Text = "";

}

else

{

if (comboBox1.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox1.Focus();

}

else

{

if (comboBox1.Text != "ערבית" && comboBox1.Text != "עברית" && comboBox1.Text != "אנגלית")

{

MessageBox.Show("הספריה מכילה ספרים רק בשפה האנגלית העברית ו הערבית ");

comboBox1.Text = "";

comboBox1.Focus();

}

else

{

if (textBox6.Text=="")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox6.Focus();

}

else

{

int num1 = Convert.ToInt32(textBox6.Text);

if (num1 > 50)

{

MessageBox.Show("הספריה לא מכילה יותר מ-50 ספר מאותו סוג ");

textBox6.Text = "";

textBox6.Focus();

}

else

{

if (comboBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox2.Focus();

}

else

{

if (comboBox2.Text != "כתבי עת" && comboBox2.Text != "סיפורים" && comboBox2.Text != "אגדות" && comboBox2.Text != "ספרי לימוד")

{

MessageBox.Show("הספריה מכילה כתבי עת,סיפורים,אגדות,ספרי לימוד ו אי אפשר להוסיף סוגי ספרים אחרים");

comboBox2.Text = "";

comboBox2.Focus();

}

else

{

if (comboBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox3.Focus();

}

else

{

if (comboBox3.Text != "0-3" && comboBox3.Text != "3-6" && comboBox3.Text != "6-10" && comboBox3.Text != "11-14" && comboBox3.Text != "15-18" && comboBox3.Text != "18 ו מעלה ")

{

MessageBox.Show(" נתון שגוי!! יש לבחור אחת מאפשריות הקיימות");

comboBox3.Text = "";

comboBox3.Focus();

}

else

{

if (comboBox4.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox4.Focus();

}

else

{

if (find.msomn(comboBox4.Text) == false)

{

MessageBox.Show("נתון שגוי");

comboBox4.Text = "";

comboBox4.Focus();

}

else

{

if (comboBox5.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox5.Focus();

}

else

{

if (find.letterr(comboBox5.Text) == false)

{

MessageBox.Show("נתון שגוי!! יש לבחור מהאפשריות הקיימות");

comboBox5.Text = "";

comboBox5.Focus();

}

else

{

bool x;

kr = liDataSet.spakim;

x = find.company(kr, textBox11.Text);

if (x == false)

{

MessageBox.Show("חברה זו לא קיימת תחילה יש להזין נתוני חברה זו");

textBox11.Focus();

}

else

{

kr3 = liDataSet.books;

int a = find.kayam(kr3, maskedTextBox1.Text);

if (a != -1)

{

int num3;

num3 = Convert.ToInt32(kr3[a]["amount"]);

int num2 = Convert.ToInt32(textBox6.Text);

num3 = num3 + num2;

string ss;

ss = Convert.ToString(num3);

int num4;

num4 = Convert.ToInt32(kr3[a]["otakim"]);

num4 = num4 + num2;

string ss2 = Convert.ToString(num4);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "',[otakim]='" + ss2 + "' where [code]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

add\_book\_Load(sender, e);

if (textBox5.Text=="z")

this.Close();

}

else

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

string ww = comboBox5.Text.ToUpper();

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into books ([code],[name],[writername],[year],[language],[amount],[type],[age],[msomn],[raw],[company],[otakim]) VALUES ('" + maskedTextBox1.Text + "','" + textBox2.Text + "','" + textBox3.Text + "','" + textBox4.Text + "','" + comboBox1.Text + "','" + textBox6.Text + "','" + comboBox2.Text + "','" + comboBox3.Text + "','" + comboBox4.Text + "','" + ww + "','" + textBox11.Text + "','" + textBox6.Text + "')";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

add\_book\_Load(sender, e);

if (textBox5.Text=="z")

{

this.Close();

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

private void add\_book\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.spakim' table. You can move, or remove it, as needed.

this.spakimTableAdapter.Fill(this.liDataSet.spakim);

//TODO: This line of code loads data into the 'liDataSet.books' table. You can move, or remove it, as needed.

this.booksTableAdapter.Fill(this.liDataSet.books);

textBox2.Text = Form1.pass;

textBox6.Text = Form1.pass3;

textBox11.Text = Form1.pass2;

textBox5.Text = Form1.pass4;

}

private void button2\_Click(object sender, EventArgs e)

{

if (textBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox2.Focus();

}

else

{

if (textBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox3.Focus();

}

else

{

if (textBox4.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox4.Focus();

}

else

{

int num = Convert.ToInt32(textBox4.Text);

if (num > 2018 || num < 1800)

{

MessageBox.Show("נתון שגוי ");

textBox4.Focus();

textBox4.Text = "";

}

else

{

if (comboBox1.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox1.Focus();

}

else

{

if (comboBox1.Text != "ערבית" && comboBox1.Text != "עברית" && comboBox1.Text != "אנגלית")

{

MessageBox.Show("הספריה מכילה ספרים רק בשפה האנגלית העברית ו הערבית ");

comboBox1.Text = "";

comboBox1.Focus();

}

else

{

if (textBox6.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox6.Focus();

}

else

{

int num1 = Convert.ToInt32(textBox6.Text);

if (num1 > 50)

{

MessageBox.Show("הספריה לא מכילה יותר מ-50 ספר מאותו סוג ");

textBox6.Text = "";

textBox6.Focus();

}

else

{

if (comboBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox2.Focus();

}

else

{

if (comboBox2.Text != "כתבי עת" && comboBox2.Text != "סיפורים" && comboBox2.Text != "אגדות" && comboBox2.Text != "ספרי לימוד")

{

MessageBox.Show("הספריה מכילה כתבי עת,סיפורים,אגדות,ספרי לימוד ו אי אפשר להוסיף סוגי ספרים אחרים");

comboBox2.Text = "";

comboBox2.Focus();

}

else

{

if (comboBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox3.Focus();

}

else

{

if (comboBox3.Text != "0-3" && comboBox3.Text != "3-6" && comboBox3.Text != "6-10" && comboBox3.Text != "11-14" && comboBox3.Text != "15-18" && comboBox3.Text != "18 ו מעלה ")

{

MessageBox.Show(" נתון שגוי!! יש לבחור אחת מאפשריות הקיימות");

comboBox3.Text = "";

comboBox3.Focus();

}

else

{

if (comboBox4.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox4.Focus();

}

else

{

if (find.msomn(comboBox4.Text) == false)

{

MessageBox.Show("נתון שגוי");

comboBox4.Text = "";

comboBox4.Focus();

}

else

{

if (comboBox5.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

comboBox5.Focus();

}

else

{

if (find.letterr(comboBox5.Text) == false)

{

MessageBox.Show("נתון שגוי!! יש לבחור מהאפשריות הקיימות");

comboBox5.Text = "";

comboBox5.Focus();

}

else

{

bool x;

kr = liDataSet.spakim;

x = find.company(kr, textBox11.Text);

if (x == false)

{

MessageBox.Show("חברה זו לא קיימת תחילה יש להזין נתוני חברה זו");

textBox11.Focus();

}

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [books] SET [name]='" + textBox2.Text + "',[writername]='" + textBox3.Text + "',[year]='" + textBox4.Text + "',[language]='" + comboBox1.Text + "',[amount]='" + textBox6.Text + "',[type]='" + comboBox2.Text + "',[age]='" + comboBox3.Text + "',[msomn]='" + comboBox4.Text + "',[raw]='" + comboBox5.Text + "',[company]='" + textBox11.Text + "',[otakim]='" + textBox6.Text + "' where [code]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

add\_book\_Load(sender, e);

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

private void textBox9\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void textBox7\_TextChanged(object sender, EventArgs e)

{

}

private void maskedTextBox1\_MaskInputRejected(object sender, MaskInputRejectedEventArgs e)

{

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

if (textBox3.Text.Length > 0)

{

string s = textBox3.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox3.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox3.Text = s1;

}

else

textBox3.Clear();

}

}

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void comboBox3\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

if (textBox6.Text.Length > 0)

{

string s = textBox6.Text;

int i = s.Length - 1;

if (!char.IsNumber(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox6.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox6.Text = s1;

}

else

textBox6.Clear();

}

}

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

if (textBox4.Text.Length > 0)

{

string s = textBox4.Text;

int i = s.Length - 1;

if (!char.IsNumber(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox4.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox4.Text = s1;

}

else

textBox4.Clear();

}

}

}

private void textBox11\_TextChanged(object sender, EventArgs e)

{

if (textBox11.Text.Length > 0)

{

string s = textBox11.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox11.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox11.Text = s1;

}

else

textBox11.Clear();

}

}

}

private void comboBox5\_SelectedIndexChanged(object sender, EventArgs e)

{

if (comboBox5.Text.Length > 0)

{

string s = comboBox5.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

comboBox5.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

comboBox5.Text = s1;

}

}

}

}

private void comboBox4\_SelectedIndexChanged(object sender, EventArgs e)

{

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

//int x = find.kayam(kr1)

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox6.Text = dataGridView1[5, e.RowIndex].Value.ToString();

textBox11.Text = dataGridView1[10, e.RowIndex].Value.ToString();

comboBox1.Text = dataGridView1[4, e.RowIndex].Value.ToString();

comboBox2.Text = dataGridView1[6, e.RowIndex].Value.ToString();

comboBox3.Text = dataGridView1[7, e.RowIndex].Value.ToString();

comboBox4.Text = dataGridView1[8, e.RowIndex].Value.ToString();

comboBox5.Text = dataGridView1[9, e.RowIndex].Value.ToString();

}

}

private void textBox12\_TextChanged(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

kr3 = liDataSet.books;

DataTable dz = new DataTable();

dz.Columns.Add("code");

dz.Columns.Add("name");

dz.Columns.Add("writername");

dz.Columns.Add("year");

dz.Columns.Add("language");

dz.Columns.Add("amount");

dz.Columns.Add("type");

dz.Columns.Add("age");

dz.Columns.Add("msomn");

dz.Columns.Add("raw");

dz.Columns.Add("company");

dz.Columns.Add("otakim");

for (int j = 0; j < kr3.Rows.Count; j++)

{

for (int k = 0; k < kr3.Columns.Count; k++)

{

if (kr3.Rows[j][k].ToString().Contains(textBox1.Text))

{

dz.Rows.Add(kr3[j][0], kr3[j][1], kr3[j][2], kr3[j][3], kr3[j][4], kr3[j][5], kr3[j][6], kr3[j][7], kr3[j][8], kr3[j][9], kr3[j][10],kr3[j][11]);

k = kr3.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void dataGridView1\_CellClick\_1(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox6.Text = dataGridView1[5, e.RowIndex].Value.ToString();

textBox11.Text = dataGridView1[10, e.RowIndex].Value.ToString();

comboBox1.Text = dataGridView1[4, e.RowIndex].Value.ToString();

comboBox2.Text = dataGridView1[6, e.RowIndex].Value.ToString();

comboBox3.Text = dataGridView1[7, e.RowIndex].Value.ToString();

comboBox4.Text = dataGridView1[8, e.RowIndex].Value.ToString();

comboBox5.Text = dataGridView1[9, e.RowIndex].Value.ToString();

}

}

}

}

public partial class bdikalhazmana : Form

{

public static string pass;

public static string pass1;

public static string pass2;

public static bool pass3;

public bdikalhazmana()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.spakimDataTable kr;

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void bdikalhazmana\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.spakim' table. You can move, or remove it, as needed.

this.spakimTableAdapter.Fill(this.liDataSet.spakim);

textBox1.Text = borrowing.pass3;

}

private void button1\_Click(object sender, EventArgs e)

{

bool x;

kr = liDataSet.spakim;

x = find.company(kr, textBox5.Text);

if (x == false)

{

MessageBox.Show("חברה זו לא קיימת תחילה יש להזין נתוני חברה זו");

textBox5.Focus();

}

else

{

MessageBox.Show("יש אפשרות להזמין את הספר");

pass = textBox5.Text;

pass1 = textBox1.Text;

pass2 = borrowing.pass4;

pass3 = true;

hazmanotli fm10 = new hazmanotli();

fm10.Show();

pass = null;

pass1 = null;

}

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class books : Form

{

public static string pass;

public books()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.booksDataTable kr;

private void books\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.books' table. You can move, or remove it, as needed.

this.booksTableAdapter.Fill(this.liDataSet.books);

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

pass = textBox2.Text;

borrowing fm7 = new borrowing();

fm7.Show();

pass = null;

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void textBox12\_TextChanged(object sender, EventArgs e)

{

kr = liDataSet.books;

DataTable dz = new DataTable();

dz.Columns.Add("code");

dz.Columns.Add("name");

dz.Columns.Add("writername");

dz.Columns.Add("year");

dz.Columns.Add("language");

dz.Columns.Add("amount");

dz.Columns.Add("type");

dz.Columns.Add("age");

dz.Columns.Add("msomn");

dz.Columns.Add("raw");

dz.Columns.Add("company");

dz.Columns.Add("otakim");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j][k].ToString().Contains(textBox12.Text))

{

dz.Rows.Add(kr[j][0], kr[j][1], kr[j][2], kr[j][3], kr[j][4], kr[j][5], kr[j][6], kr[j][7], kr[j][8], kr[j][9], kr[j][10],kr[j][11]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

}

private void label12\_Click(object sender, EventArgs e)

{

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void dataGridView1\_CellClick\_1(object sender, DataGridViewCellEventArgs e)

{

textBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox5.Text = dataGridView1[4, e.RowIndex].Value.ToString();

textBox6.Text = dataGridView1[5, e.RowIndex].Value.ToString();

textBox7.Text = dataGridView1[6, e.RowIndex].Value.ToString();

textBox8.Text = dataGridView1[7, e.RowIndex].Value.ToString();

textBox9.Text = dataGridView1[8, e.RowIndex].Value.ToString();

textBox10.Text = dataGridView1[9, e.RowIndex].Value.ToString();

textBox11.Text = dataGridView1[10, e.RowIndex].Value.ToString();

}

private void dataGridView1\_CellClick\_2(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

textBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox5.Text = dataGridView1[4, e.RowIndex].Value.ToString();

textBox6.Text = dataGridView1[5, e.RowIndex].Value.ToString();

textBox7.Text = dataGridView1[6, e.RowIndex].Value.ToString();

textBox8.Text = dataGridView1[7, e.RowIndex].Value.ToString();

textBox9.Text = dataGridView1[8, e.RowIndex].Value.ToString();

textBox10.Text = dataGridView1[9, e.RowIndex].Value.ToString();

textBox11.Text = dataGridView1[10, e.RowIndex].Value.ToString();

textBox13.Text = dataGridView1[11, e.RowIndex].Value.ToString();

}

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

}

private void textBox13\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class borrowing : Form

{

public static string pass;

public static string pass2;

public static string pass3;

public static string pass4;

public borrowing()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.borrowDataTable kr;

liDataSet.kartis\_korehDataTable kr1;

liDataSet.booksDataTable kr2;

liDataSet.hazmanotDataTable kr3;

liDataSet.trashDataTable kr4;

private void borrowing\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.trash' table. You can move, or remove it, as needed.

this.trashTableAdapter.Fill(this.liDataSet.trash);

// TODO: This line of code loads data into the 'liDataSet.hazmanot' table. You can move, or remove it, as needed.

this.hazmanotTableAdapter.Fill(this.liDataSet.hazmanot);

// TODO: This line of code loads data into the 'liDataSet.books' table. You can move, or remove it, as needed.

this.booksTableAdapter.Fill(this.liDataSet.books);

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

// TODO: This line of code loads data into the 'liDataSet.borrow' table. You can move, or remove it, as needed.

this.borrowTableAdapter.Fill(this.liDataSet.borrow);

textBox5.Text = books.pass;

}

private void books\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.books' table. You can move, or remove it, as needed.

this.booksTableAdapter.Fill(this.liDataSet.books);

}

private void button1\_Click(object sender, EventArgs e)

{

////string sss = find.date(dateTimePicker1.Text);

//MessageBox.Show(sss);

int x;

kr2 = liDataSet.books;

if (textBox5.Text == "")

{

MessageBox.Show("אי אפשר לבצע השאלה בלי הזנת שם הספר");

textBox5.Focus();

}

else

{

x = find.yes(kr2, textBox5.Text);

if (x == -1)

{

MessageBox.Show("ספר זה לא קיים בספריה");

DialogResult aa = MessageBox.Show(" אנא הקש אישור אם ברצונכה לבדוק אפשרות הזמנת הספר ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

pass3 = textBox5.Text;

pass4 = maskedTextBox1.Text;

bdikalhazmana fm8 = new bdikalhazmana();

fm8.Show();

pass3 = null;

//pass4 = null;

this.Close();

}

textBox5.Text = "";

textBox5.Focus();

}

else

{

if (kr2[x]["amount"].ToString() == "0")

{

DialogResult aa = MessageBox.Show(" ספר זה לא זמין ,לחץ אישור להזמנת ספר זה ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

pass = maskedTextBox1.Text;

pass2 = textBox5.Text;

hazmanot fm4 = new hazmanot();

fm4.Show();

pass = null;

pass2 = null;

}

else

{

textBox5.Text = "";

textBox5.Focus();

}

}

else

{

if (kr2[x]["msomn"].ToString() == "yes")

{

MessageBox.Show("ספר זה מסומן!! אי אפשר להשאיל אותו");

textBox5.Text = "";

textBox5.Focus();

}

else

{

bool b11;

b11 = find.TowTimes(kr, maskedTextBox1.Text);

if (b11 == true)

{

MessageBox.Show("אי אשפר לבצע השאלה!! לקוח זה יש לו שתי השאלות");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

DialogResult aa = MessageBox.Show(" האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into borrow ([id],[book name],[date of borowing]) VALUES ('" + maskedTextBox1.Text + "','" + textBox5.Text + "','" + dateTimePicker1.Value.ToShortDateString() + "')";

con.Open();

cmd.Connection = con;

cmd.ExecuteNonQuery();

int num;

num = Convert.ToInt32(kr2[x]["amount"]);

num = num - 1;

string ss;

ss = Convert.ToString(num);

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "' where [name]='" + textBox5.Text + "'";

cmd.ExecuteNonQuery();

con.Close();

books\_Load(sender, e);

borrowing\_Load(sender, e);

}

}

}

}

}

}

}

private void button2\_Click(object sender, EventArgs e)

{

kr2 = liDataSet.books;

kr = liDataSet.borrow;

string s;

s = dateTimePicker1.Text;

int x;

x = find.yes(kr2, textBox5.Text);

int y = find.findborrow(kr, maskedTextBox1.Text);

string s2 = kr[y]["book name"].ToString();

int z = find.yes(kr2, s2);

if (x == -1)

{

MessageBox.Show("ספר זה לא קיים");

textBox5.Text = "";

textBox5.Focus();

}

else if (kr2[x]["amount"].ToString() == "0")

{

MessageBox.Show("ספר זה לא זמין כעת");

}

else if (kr2[x]["msomn"].ToString() == "yes")

{

MessageBox.Show("ספר זה מסומן!! אי אפשר להשאיל אותו");

textBox5.Text = "";

textBox5.Focus();

}

else if (find.same(kr, maskedTextBox1.Text, textBox5.Text) == true)

{

MessageBox.Show("אין באפשרותך לשנות תאריך בגלל שלא שיניתה את הספר");

}

else

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [borrow] SET [book name]='" + textBox5.Text + "',[date of borowing]='" + textBox1.Text + "' where [id]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

int num;

num = Convert.ToInt32(kr2[x]["amount"]);

num = num - 1;

string ss;

ss = Convert.ToString(num);

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "'where[name]='" + textBox5.Text + "'";

cmd.ExecuteNonQuery();

int num2;

num2 = Convert.ToInt32(kr2[z]["amount"]);

num = num2 + 1;

ss = Convert.ToString(num);

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "'where[name]='" + s2 + "'";

cmd.ExecuteNonQuery();

con.Close();

books\_Load(sender, e);

borrowing\_Load(sender, e);

maskedTextBox1.Text = "";

textBox5.Text = "";

textBox7.Text = "";

textBox1.Text = "";

}

}

}

private void textBox7\_TextChanged(object sender, EventArgs e)

{

kr = liDataSet.borrow;

DataTable dz = new DataTable();

dz.Columns.Add("id");

dz.Columns.Add("book name");

dz.Columns.Add("date of borowing");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j][k].ToString().Contains(textBox7.Text))

{

dz.Rows.Add(kr[j][0], kr[j][1], kr[j][2]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

private void button3\_Click(object sender, EventArgs e)

{

int x;

kr2 = liDataSet.books;

kr1 = liDataSet.kartis\_koreh;

x = find.yes(kr2, textBox5.Text);

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

kr3 = liDataSet.hazmanot;

if (aa == DialogResult.Yes)

{

string y = find.hazmanamokhana(kr3, textBox5.Text);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "delete from borrow where [id]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

int num;

num = Convert.ToInt32(kr2[x]["amount"]);

num = num + 1;

string ss;

ss = Convert.ToString(num);

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "'where[name]='" + textBox5.Text + "'";

cmd.ExecuteNonQuery();

textBox5.Text = "";

textBox7.Text = "";

cmd.CommandText = "UPDATE [kartis koreh] SET [hayav]='" + "לא" + "' where [id]='" + maskedTextBox1.Text + "'";

maskedTextBox1.Text = "";

textBox1.Text = "";

cmd.ExecuteNonQuery();

if (y != "")

{

int z = find.IsFound(kr1, y);

string tellephone = kr1[z]["phone"].ToString();

MessageBox.Show(tellephone + "התקשר למספר ו תודיע לו שההזמנה מוכנה");

}

cmd.CommandText = "UPDATE [hazmanot] SET [messagedate]='" + dateTimePicker1.Value.ToShortDateString() + "'where[id]='" + y + "'";

cmd.ExecuteNonQuery();

con.Close();

borrowing\_Load(sender, e);

books\_Load(sender, e);

card\_Load(sender, e);

hazmanot\_Load(sender, e);

}

}

private void hazmanot\_Load(object sender, EventArgs e)

{

}

private void card\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

MessageBox.Show(dateTimePicker1.Text);

}

private void maskedTextBox1\_MaskInputRejected(object sender, MaskInputRejectedEventArgs e)

{

}

private void button6\_Click(object sender, EventArgs e)

{

textBox1.Visible = false;

button4.Visible = true;

label1.Visible = true;

label8.Visible = false;

textBox7.Visible = false;

button2.Visible = false;

button3.Visible = false;

label5.Visible = false;

label6.Visible = false;

textBox5.Visible = false;

dateTimePicker1.Visible = false;

maskedTextBox1.Visible = true;

maskedTextBox1.Text = "";

maskedTextBox1.Enabled = true;

MessageBox.Show("הקש ת.ז קורא");

maskedTextBox1.Focus();

}

private void button5\_Click(object sender, EventArgs e)

{

label1.Visible = true;

textBox1.Visible = true;

label5.Visible = true;

label6.Visible = true;

label8.Visible = true;

button2.Visible = true;

button3.Visible = true;

textBox7.Visible = true;

button9.Visible = true;

textBox7.Text = "";

maskedTextBox1.Visible = true;

maskedTextBox1.Text = "";

textBox5.Visible = true;

textBox5.Text = "";

dateTimePicker1.Visible = true;

dateTimePicker1.Text = "";

button1.Visible = false;

button4.Visible = false;

maskedTextBox1.Enabled = false;

}

private void button4\_Click\_1(object sender, EventArgs e)

{

kr1 = liDataSet.kartis\_koreh;

kr = liDataSet.borrow;

if (maskedTextBox1.Text.Length != 9 || maskedTextBox1.Text.IndexOf(" ") > -1)

{

MessageBox.Show("נתון שהיזנתה אינו חוקי ");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

int x = find.IsFound(kr1, maskedTextBox1.Text);

if (x != -1)

{

bool b;

b = find.TowTimes(kr, maskedTextBox1.Text);

if (b == true)

{

MessageBox.Show("אי אשפר לבצע השאלה!! לקוח זה יש לו השאלה קיימת ");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

textBox1.Visible = false;

label5.Visible = true;

label6.Visible = false;

textBox5.Visible = true;

dateTimePicker1.Visible = true;

button1.Visible = true;

MessageBox.Show(x.ToString("הקש את שם הספר"));

textBox5.Focus();

button4.Visible = false;

}

}

else

{

MessageBox.Show("לקוח זה לא קיים");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

}

}

private void button7\_Click(object sender, EventArgs e)

{

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void maskedTextBox1\_MaskInputRejected\_1(object sender, MaskInputRejectedEventArgs e)

{

}

//private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

//{

// maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

// textBox5.Text = dataGridView1[4, e.RowIndex].Value.ToString();

// dateTimePicker1.Text = dataGridView1[5, e.RowIndex].Value.ToString();

//}

private void label5\_Click(object sender, EventArgs e)

{

}

private void dataGridView1\_CellClick\_1(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox5.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox1.Text = dataGridView1[2, e.RowIndex].Value.ToString();

}

}

private void button8\_Click(object sender, EventArgs e)

{

textBox5.Text = "";

this.Close();

}

private void label6\_Click(object sender, EventArgs e)

{

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void button9\_Click(object sender, EventArgs e)

{

int x;

kr2 = liDataSet.books;

kr1 = liDataSet.kartis\_koreh;

kr4 = liDataSet.trash;

x = find.yes(kr2, textBox5.Text);

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

kr3 = liDataSet.hazmanot;

if (aa == DialogResult.Yes)

{

string y = find.hazmanamokhana(kr3, textBox5.Text);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "delete from borrow where [id]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

int num;

num = Convert.ToInt32(kr2[x]["amount"]);

num = num + 1;

string ss;

ss = Convert.ToString(num);

cmd.CommandText = "UPDATE [books] SET [amount]='" + ss + "'where[name]='" + textBox5.Text + "'";

cmd.ExecuteNonQuery();

textBox5.Text = "";

textBox7.Text = "";

cmd.CommandText = "UPDATE [kartis koreh] SET [hayav]='" + "לא" + "' where [id]='" + maskedTextBox1.Text + "'";

maskedTextBox1.Text = "";

textBox1.Text = "";

cmd.ExecuteNonQuery();

if (y != "")

{

int z = find.IsFound(kr1, y);

string tellephone = kr1[z]["phone"].ToString();

MessageBox.Show(tellephone + "התקשר למספר ו תודיע לו שההזמנה מוכנה");

}

cmd.CommandText = "UPDATE [hazmanot] SET [messagedate]='" + dateTimePicker1.Value.ToShortDateString() + "'where[id]='" + y + "'";

cmd.ExecuteNonQuery();

con.Close();

borrowing\_Load(sender, e);

books\_Load(sender, e);

card\_Load(sender, e);

hazmanot\_Load(sender, e);

trash\_Load( sender, e);

}

}

private void trash\_Load(object sender, EventArgs e)

{

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class btokef : Form

{

public btokef()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.kartis\_korehDataTable kr;

private void btokef\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

kr = liDataSet.kartis\_koreh;

DataTable dz = new DataTable();

dz.Columns.Add("id ");

dz.Columns.Add("name ");

dz.Columns.Add("last name");

dz.Columns.Add("phone");

dz.Columns.Add("start day");

dz.Columns.Add("end day");

for (int j = 0; j < kr.Rows.Count; j++)

{

if (kr.Rows[j]["tokef"].ToString() == "כו")

{

dz.Rows.Add(kr[j][0], kr[j][1], kr[j][2], kr[j][3], kr[j][4], kr[j][5]);

}

}

dataGridView1.DataSource = dz;

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class card : Form

{

public card()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.kartis\_korehDataTable kr;

liDataSet.trashDataTable kr1;

private void button2\_Click(object sender, EventArgs e)

{

string s4="";

if (checkBox2.Checked)

{

s4 = dateTimePicker3.Value.AddYears(1).ToShortDateString();

}

if (checkBox1.Checked)

{

s4 = dateTimePicker3.Value.AddMonths(6).ToShortDateString();

}

if (maskedTextBox1.Text.Length != 9 || maskedTextBox1.Text.IndexOf(" ") > -1)

{

MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו ");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

kr = liDataSet.kartis\_koreh;

if (find.IsFound(kr, maskedTextBox1.Text) != -1)

{

MessageBox.Show("לקוח קיים ");

maskedTextBox1.Text = "";

}

else

{

if (textBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox2.Focus();

}

else

{

if (textBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox3.Focus();

}

else

{

if (find.tell(maskedTextBox2.Text))

{

MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו");

maskedTextBox2.Text = "";

maskedTextBox2.Focus();

}

else

{

if (checkBox2.Checked == false && checkBox1.Checked == false)

{

MessageBox.Show("יש לבחור את תוקף המנוי");

}

else

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into [kartis koreh] ([id],[name],[last name],[phone],[start day],[end day],[hayav],[tokef]) VALUES ('" + maskedTextBox1.Text + "','" + textBox2.Text + "','" + textBox3.Text + "','" + maskedTextBox2.Text + "','" + dateTimePicker3.Value + "','" + dateTimePicker3.Value.AddYears(1).ToShortDateString() + "','" + "לא" + "','"+"כו"+"')";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

card\_Load(sender, e);

}

}

}

}

}

}

}

}

private void card\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.trash' table. You can move, or remove it, as needed.

this.trashTableAdapter.Fill(this.liDataSet.trash);

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

}

private void button5\_Click(object sender, EventArgs e)

{

string s4 = "";

if (checkBox2.Checked)

{

s4 = dateTimePicker3.Value.AddYears(1).ToShortDateString();

}

if (checkBox1.Checked)

{

s4 = dateTimePicker3.Value.AddMonths(6).ToShortDateString();

}

if (maskedTextBox1.Text.Length != 9 || maskedTextBox1.Text.IndexOf(" ") > -1)

{

MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו ");

maskedTextBox1.Text = "";

maskedTextBox1.Focus();

}

else

{

if (textBox2.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox2.Focus();

}

else

{

if (textBox3.Text == "")

{

MessageBox.Show("ישנם נתונים חסרים");

textBox3.Focus();

}

else

{

//if (maskedTextBox2.Text.Length != 11)

//{

// MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו ");

// maskedTextBox2.Text = "";

// maskedTextBox2.Focus();

//}

//else

{

string s2 = maskedTextBox2.Text;

int count = 0;

// for (int i = 0; i < s.Length;i++ )

//{

// if(s2[i]=='')

// {

// count++;

// }

//}

if (count > 1)

{

MessageBox.Show("נתון שהזנה שגוי !!יש לתקן אותו");

maskedTextBox2.Text = "";

maskedTextBox2.Focus();

}

else

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [kartis koreh] SET [name]='" + textBox2.Text + "',[last name]='" + textBox3.Text + "',[phone]='" + maskedTextBox2.Text + "',[start day]='" + dateTimePicker3.Value + "',[end day]='" + dateTimePicker3.Value.AddYears(1).ToShortDateString() + "' where [id]='" + maskedTextBox1.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

card\_Load(sender, e);

}

}

}

}

}

}

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

}

private void button4\_Click(object sender, EventArgs e)

{

}

private void textBox7\_TextChanged(object sender, EventArgs e)

{

kr = liDataSet.kartis\_koreh;

DataTable dz = new DataTable();

dz.Columns.Add("id");

dz.Columns.Add("name");

dz.Columns.Add("last name");

dz.Columns.Add("phone");

dz.Columns.Add("start day");

dz.Columns.Add("end day");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j][k].ToString().Contains(textBox7.Text))

{

dz.Rows.Add(kr[j][0], kr[j][1], kr[j][2], kr[j][3], kr[j][4], kr[j][5]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

private void button6\_Click(object sender, EventArgs e)

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into [trash] ([bitol],[id botal],[book],[dateb]) VALUES ('" + "k" + "','" + maskedTextBox1.Text + "','" + "" + "','" + dateTimePicker3.Value + "')";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

cmd.CommandText = "delete from [kartis koreh] where [id]='" + maskedTextBox1.Text + "'";

cmd.ExecuteNonQuery();

con.Close();

card\_Load(sender, e);

maskedTextBox1.Text = "";

maskedTextBox2.Text = "";

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

textBox4.Text = "";

}

}

private void button3\_Click(object sender, EventArgs e)

{

textBox1.Visible = true;

textBox4.Visible = true;

label1.Visible = true;

label2.Visible = true;

label3.Visible = true;

label4.Visible = true;

label5.Visible = true;

label6.Visible = true;

maskedTextBox1.Visible = true;

textBox2.Visible = true;

textBox3.Visible = true;

maskedTextBox2.Visible = true;

textBox7.Visible = true;

button5.Visible = true;

button6.Visible = true;

label7.Visible = true;

checkBox2.Visible = false;

checkBox1.Visible = false;

label9.Visible = true;

textBox5.Visible = true;

}

private void button1\_Click(object sender, EventArgs e)

{

textBox1.Visible = false;

textBox4.Visible = false;

label1.Visible = true;

label2.Visible = true;

label3.Visible = true;

label4.Visible = true;

label5.Visible = false;

label6.Visible = false;

maskedTextBox1.Visible = true;

maskedTextBox1.Text = "";

textBox2.Visible = true;

textBox3.Visible = true;

maskedTextBox2.Visible = true;

button2.Visible = true;

button6.Visible = false;

button5.Visible = false;

label7.Visible = false;

textBox7.Visible = false;

checkBox2.Visible = true;

checkBox1.Visible = true;

textBox5.Visible = false;

label9.Visible = false;

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

if (textBox2.Text.Length > 0)

{

string s = textBox2.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox2.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox2.Text = s1;

}

else

textBox2.Clear();

}

}

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

if (textBox3.Text.Length > 0)

{

string s = textBox3.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox3.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox3.Text = s1;

}

else

textBox3.Clear();

}

}

}

private void maskedTextBox2\_MaskInputRejected(object sender, MaskInputRejectedEventArgs e)

{

}

private void maskedTextBox1\_MaskInputRejected(object sender, MaskInputRejectedEventArgs e)

{

}

private void dateTimePicker2\_ValueChanged(object sender, EventArgs e)

{

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

maskedTextBox2.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox1.Text= dataGridView1[4, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[5, e.RowIndex].Value.ToString();

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox1.Checked)

{

checkBox2.Checked = false;

}

}

private void checkBox2\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox2.Checked)

{

checkBox1.Checked = false;

}

}

private void button4\_Click\_1(object sender, EventArgs e)

{

////string s = find.date(dateTimePicker1.Text);

////MessageBox.Show(s);

//string s2 = find.datemonth(dateTimePicker1.Text);

//MessageBox.Show(s2);

//string s3 = find.changemonth(s2, dateTimePicker1.Text);

//MessageBox.Show(s3);

//if (find.tell(maskedTextBox2.Text))

//{

// MessageBox.Show("ok");

//}

//else

kr = liDataSet.kartis\_koreh;

Form2 fm1 = new Form2();

fm1.Show();

int x = kr.Rows.Count;

string s = Convert.ToString(x);

MessageBox.Show(s);

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void dataGridView1\_CellClick\_1(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

maskedTextBox1.Text = dataGridView1[0, e.RowIndex].Value.ToString();

maskedTextBox2.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox3.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox1.Text = dataGridView1[4, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[5, e.RowIndex].Value.ToString();

textBox5.Text = dataGridView1[6, e.RowIndex].Value.ToString();

}

}

private void dataGridView1\_CellContentClick\_2(object sender, DataGridViewCellEventArgs e)

{

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class Form1 : Form

{

public static string pass;

public static string pass2;

public static string pass3;

public static string pass4;

public static Stack<int> pass5;

public Form1()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.borrowDataTable kr;

liDataSet.kartis\_korehDataTable kr2;

liDataSet.hazmanotDataTable kr3;

//liDataSet.booksDataTable kr4;

liDataSet.hazmanotlibraryDataTable kr5;

private void button2\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

//dohot fm = new dohot();

//fm.Show();

}

private void button2\_Click\_1(object sender, EventArgs e)

{

borrowing fm1 = new borrowing();

fm1.Show();

}

private void button3\_Click(object sender, EventArgs e)

{

pass4 = "l";

add\_book fm2 = new add\_book();

fm2.Show();

}

private void button4\_Click(object sender, EventArgs e)

{

card fm3 = new card();

fm3.Show();

}

//private void button5\_Click(object sender, EventArgs e)

//{

// hazmanot fm4=new hazmanot

// fm4 .Show();

private void חדשעלהמדףToolStripMenuItem\_Click(object sender, EventArgs e)

{

new\_in fm30 = new new\_in();

fm30.Show();

}

private void נוערToolStripMenuItem\_Click(object sender, EventArgs e)

{

}

private void השאלהמהספריהToolStripMenuItem\_Click(object sender, EventArgs e)

{

borrowing fm6 = new borrowing();

fm6.Show();

}

private void toolStripComboBox1\_Click(object sender, EventArgs e)

{

}

private void הכלToolStripMenuItem\_Click(object sender, EventArgs e)

{

legends fm23 = new legends();

fm23.Show();

}

private void button5\_Click\_1(object sender, EventArgs e)

{

hazmanot fm4 = new hazmanot();

fm4.Show();

}

private void צעיריםToolStripMenuItem\_Click(object sender, EventArgs e)

{

}

private void הכלToolStripMenuItem2\_Click(object sender, EventArgs e)

{

books fm5 = new books();

fm5.Show();

}

private void לקוחותשחייביםלשלםToolStripMenuItem\_Click(object sender, EventArgs e)

{

//dohot fm6 = new dohot();

//fm6.Show();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

tkanon fm9 = new tkanon();

fm9.Show();

}

private void דוחספריםToolStripMenuItem\_Click(object sender, EventArgs e)

{

}

private void ספריםקייToolStripMenuItem\_Click(object sender, EventArgs e)

{

avialbe\_books fm10 = new avialbe\_books();

fm10.Show();

}

private void ספריםלאזמיניםToolStripMenuItem\_Click(object sender, EventArgs e)

{

not\_availbe\_books fm11 = new not\_availbe\_books();

fm11.Show();

}

private void חייביםלשלםToolStripMenuItem\_Click(object sender, EventArgs e)

{

hiaveem fm12 = new hiaveem();

fm12.Show();

}

private void לאהחזירוToolStripMenuItem\_Click(object sender, EventArgs e)

{

no\_reback fm12 = new no\_reback();

fm12.Show();

}

private void button6\_Click(object sender, EventArgs e)

{

}

private void button6\_Click\_1(object sender, EventArgs e)

{

Form2 fm1 = new Form2();

fm1.Show();

}

private void Form1\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.hazmanotlibrary' table. You can move, or remove it, as needed.

this.hazmanotlibraryTableAdapter.Fill(this.liDataSet.hazmanotlibrary);

// TODO: This line of code loads data into the 'liDataSet.hazmanot' table. You can move, or remove it, as needed.

this.hazmanotTableAdapter.Fill(this.liDataSet.hazmanot);

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

// TODO: This line of code loads data into the 'liDataSet.borrow' table. You can move, or remove it, as needed.

this.borrowTableAdapter.Fill(this.liDataSet.borrow);

}

private void hazmanot\_Load(object sender, EventArgs e)

{

}

private void menuStrip1\_ItemClicked(object sender, ToolStripItemClickedEventArgs e)

{

}

private void dateTimePicker1\_ValueChanged(object sender, EventArgs e)

{

}

private void card\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

}

private void button7\_Click(object sender, EventArgs e)

{

hazmanotli fm17 = new hazmanotli();

fm17.Show();

}

private void ספריםזמיניםToolStripMenuItem\_Click(object sender, EventArgs e)

{

avialbe\_books fm18 = new avialbe\_books();

fm18.Show();

}

private void ספריםלאזמיניםToolStripMenuItem1\_Click(object sender, EventArgs e)

{

not\_availbe\_books fm20 = new not\_availbe\_books();

fm20.Show();

}

private void ילדיםונוערToolStripMenuItem\_Click(object sender, EventArgs e)

{

edubooks fm21 = new edubooks();

fm21.Show();

}

private void מבוגריםToolStripMenuItem\_Click(object sender, EventArgs e)

{

stories fm22 = new stories();

fm22.Show();

}

private void השאלותקיימותToolStripMenuItem\_Click(object sender, EventArgs e)

{

hazmanot\_kayamot fm24 = new hazmanot\_kayamot();

fm24.Show();

}

private void הזמנותקיימותToolStripMenuItem\_Click(object sender, EventArgs e)

{

ashalot\_kayamot fm25 = new ashalot\_kayamot();

fm25.Show();

}

private void button8\_Click(object sender, EventArgs e)

{

kr2 = liDataSet.kartis\_koreh;

kr = liDataSet.borrow;

kr5 = liDataSet.hazmanotlibrary;

kr3=liDataSet.hazmanot;

string ehor = dateTimePicker1.Value.AddDays(-3).ToShortDateString().ToString();

for (int l = 0; l < kr3.Rows.Count; l++)

{

MessageBox.Show("hellow");

if (kr3[l]["messagedate"].ToString() == ehor)

{

string tz = kr3[l]["id"].ToString();

MessageBox.Show(tz);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "delete from hazmanot where [id]='" + tz + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

hazmanot\_Load(sender, e);

}

}

string s = dateTimePicker1.Value.AddMonths(-1).ToShortDateString().ToString();

string s2 = dateTimePicker1.Value.AddMonths(-1).AddDays(-1).ToShortDateString().ToString();

kr3 = liDataSet.hazmanot;

Stack<string> sm = new Stack<string>();

Stack<string> sm2 = new Stack<string>();

Stack<string> sm3 = new Stack<string>();

Stack<string> sm4 = new Stack<string>();

for (int i = 0; i < kr2.Rows.Count; i++)

{

string phone;

if (kr2[i]["end day"].ToString()==dateTimePicker1.Value.ToShortDateString().ToString())

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [kartis koreh] SET[tokef]='" + "לא"+ "' where [id]='" + kr2[i]["id"].ToString() + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

card\_Load(sender, e);

phone = kr2[i]["phone"].ToString();

MessageBox.Show("תתקשר ללקוח ו תודיע לו שהמנוי שלו הסתיים");

}

}

for (int i = 0; i < kr.Rows.Count; i++)

{

if (s == kr[i]["date of borowing"].ToString())

{

if (kr[i]["id"].ToString() != null)

{

sm.Push(kr[i]["id"].ToString());

sm3.Push(kr[i]["id"].ToString());

}

}

if (s2 == kr[i]["date of borowing"].ToString())

{

if (kr[i]["id"].ToString() != null)

{

sm2.Push(kr[i]["id"].ToString());

sm4.Push(kr[i]["id"].ToString());

}

}

}

while (sm.Count != 0)

{

int i = find.IsFound(kr2, sm.Pop());

string sss = kr2[i]["phone"].ToString();

if (kr2[i]["hayav"].ToString() == "לא")

{

MessageBox.Show(" תזכורת ללקוח שזמן הההשאלה הסתיים אנא התקששר למספר " + sss);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [kartis koreh] SET [hayav]='" + "\*\*" + "' where [id]='" + kr2[i]["id"].ToString() + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

}

}

card\_Load(sender, e);

if (sm2.Count != 0)

{

MessageBox.Show("לקוחות שיופיעו קיבלו הודעה ו לא החזירו את הספרים");

while (sm2.Count != 0)

{

int i = find.IsFound(kr2, sm2.Pop());

string sss2 = kr2[i]["phone"].ToString();

MessageBox.Show(" אנא הודיע ללקוח שקיבל כנס " + sss2);

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [kartis koreh] SET [hayav]='" + "200 ש''ח" + "' where [id]='" + kr2[i]["id"].ToString() + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

card\_Load(sender, e);

}

}

if (sm3.Count == 0 && sm4.Count == 0)

{

MessageBox.Show("ok");

}

Stack<int> index123 = new Stack<int>();

Stack<int> index456 = new Stack<int>();

string book;

string kbala = dateTimePicker1.Value.AddDays(0).ToShortDateString().ToString();

for (int j = 0; j < kr5.Rows.Count; j++)

{

if (kbala == kr5[j]["date"].ToString())

{

book = kr5[j]["bookname"].ToString();

DialogResult aa = MessageBox.Show( book+" האם ההזמנה הגיעה" , "exit", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

index123.Push(j);

index456.Push(j);

}

else

MessageBox.Show("ההזמנה לא הגיעה");

}

}

int x = -1;

string y;

while (index456.Count != 0)

{

x = index456.Peek();

y = find.hazmanamokhana(kr3, kr5[x]["bookname"].ToString());

if (y != "")

{

int z = find.IsFound(kr2, y);

string tellephone = kr2[z]["phone"].ToString();

MessageBox.Show(tellephone + " " + "התקשר למספר ו תודיע לו שההזמנה מוכנה");

x = index456.Pop();

}

OleDbCommand cmd = new OleDbCommand();

cmd.Connection = con;

con.Open();

cmd.CommandText = "UPDATE [hazmanot] SET [messagedate]='" + dateTimePicker1.Value.ToShortDateString() + "'where[id]='" + y + "'";

cmd.ExecuteNonQuery();

con.Close();

hazmanot\_Load(sender, e);

}

if (index123.Count != 0)

{

x = index123.Pop();

pass = kr5[x]["bookname"].ToString();

pass2 = kr5[x]["companyname"].ToString();

pass3 = kr5[x]["kamot"].ToString();

pass4 = "z";

pass5 = index456;

add\_book fm18 = new add\_book();

fm18.Show();

pass = null;

pass2 = null;

pass3 = null;

while (index123.Count != 0)

{

pass = kr5[x]["bookname"].ToString();

pass2 = kr5[x]["companyname"].ToString();

pass3 = kr5[x]["kamot"].ToString();

pass4 = "z";

pass5 = index456;

add\_book fm17 = new add\_book();

fm17.Show();

pass = null;

pass2 = null;

pass3 = null;

x = index123.Pop();

}

}

MessageBox.Show("ok");

menuStrip1.Enabled = true;

button1.Enabled = true;

button2.Enabled = true;

button3.Enabled = true;

button4.Enabled = true;

button5.Enabled = true;

button6.Enabled = true;

button7.Enabled = true;

button8.Visible = false;

}

private void button9\_Click(object sender, EventArgs e)

{

trash fm30 = new trash();

fm30.Show();

}

private void השאלותשבוטלוToolStripMenuItem\_Click(object sender, EventArgs e)

{

borrowcan fm0 = new borrowcan();

fm0.Show();

}

private void הזמנותהסתיימוToolStripMenuItem\_Click(object sender, EventArgs e)

{

hazmanotcan fm00 = new hazmanotcan();

fm00.Show();

}

private void כרטיסיםלאבתוקףToolStripMenuItem\_Click(object sender, EventArgs e)

{

lobtokef fmm = new lobtokef();

fmm.Show();

}

private void כרטיסיםבתוקףToolStripMenuItem\_Click(object sender, EventArgs e)

{

btokef lm = new btokef();

lm.Show();

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace liii

{

public partial class Form2 : Form

{

public static bool pass;

public Form2()

{

InitializeComponent();

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

private void Form2\_Load(object sender, EventArgs e)

{

textBox2.Text = "Username";

textBox1.Text = "Password";

this.AcceptButton = button1;

}

private void pictureBox1\_Click\_1(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

if (textBox1.Text != "Password")

{

textBox1.UseSystemPasswordChar = true;

}

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void textBox2\_Click(object sender, EventArgs e)

{

textBox2.Text = "";

}

private void label2\_Click(object sender, EventArgs e)

{

}

private void textBox1\_Click(object sender, EventArgs e)

{

textBox1.Text = "";

}

private void button1\_Click(object sender, EventArgs e)

{

label3.Visible = false;

label2.Visible = false;

if (textBox2.Text=="adminadmin")

{

if (textBox1.Text=="jdlibrary2018")

{

pass = true;

spakim fm13 = new spakim();

fm13.Show();

pass = false;

this.Close();

}

else

{

label3.Visible = true;

}

}

else if (textBox1.Text != "jdlibrary2018")

{

label2.Visible = true;

label3.Visible = true;

}

else

{

label2.Visible = true;

}

}

private void button2\_Click(object sender, EventArgs e)

{

spakim fm13 = new spakim();

fm13.Show();

this.Close();

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class hazmanotli : Form

{

public static string pass1;

public hazmanotli()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.spakimDataTable kr;

liDataSet.hazmanotlibraryDataTable kr1;

private void hazmanotli\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.spakim' table. You can move, or remove it, as needed.

this.spakimTableAdapter.Fill(this.liDataSet.spakim);

// TODO: This line of code loads data into the 'liDataSet.hazmanotlibrary' table. You can move, or remove it, as needed.

this.hazmanotlibraryTableAdapter.Fill(this.liDataSet.hazmanotlibrary);

textBox1.Text = bdikalhazmana.pass;

textBox5.Text = bdikalhazmana.pass1;

if ( bdikalhazmana.pass3==true)

{

textBox3.Enabled = false;

}

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox5.Text == "")

{

MessageBox.Show("אי אפשר לבצע השאלה בלי הזנת שם הספר");

textBox5.Focus();

}

else

{

if (textBox1.Text=="")

{

MessageBox.Show("אי אפשר לבצע השאלה בלי הזנת שם הספר");

textBox1.Focus();

}

else

{

if (textBox3.Text == "" && textBox3.Enabled==true)

{

MessageBox.Show("אי אפשר לבצע השאלה בלי הזנת שם הספר");

textBox3.Focus();

}

else

{

bool x;

kr = liDataSet.spakim;

x = find.company(kr, textBox1.Text);

if (x == false)

{

MessageBox.Show("חברה זו לא קיימת תחילה יש להזין נתוני חברה זו");

textBox1.Focus();

}

else

{

int f = find.numsapak(kr, textBox1.Text);

string s = kr[f]["cophone"].ToString();

DialogResult aa = MessageBox.Show("התקשר לחברה ו אמת הזמנה "+s, "exit", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

string s1;

if (textBox3.Enabled==false)

{

s1 = "1";

}

else

s1 = textBox3.Text;

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into hazmanotlibrary ([bookname],[companyname],[date],[kamot]) VALUES ('" + textBox5.Text + "','" + textBox1.Text + "','" + dateTimePicker1.Value.ToShortDateString() + "','" +s1 + "')";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

cmd.CommandText = "INSERT into hazmanot ([id],[book name],[date of hazmana]) VALUES ('" + bdikalhazmana.pass2 + "','" + textBox5.Text + "','" + dateTimePicker1.Value.ToShortDateString() + "')";

cmd.ExecuteNonQuery();

con.Close();

hazmanotli\_Load(sender, e);

hazmanot\_Load(sender, e);

textBox2.Text = "";

textBox3.Text = "";

}

}

}

}

}

}

private void hazmanot\_Load(object sender, EventArgs e)

{

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

if (textBox3.Text.Length > 0)

{

string s = textBox3.Text;

int i = s.Length - 1;

if (!char.IsNumber(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox3.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox3.Text = s1;

}

else

textBox3.Clear();

}

}

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

if (textBox1.Text.Length > 0)

{

string s = textBox1.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox1.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox1.Text = s1;

}

else

textBox1.Clear();

}

}

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

kr1 = liDataSet.hazmanotlibrary;

DataTable dz = new DataTable();

dz.Columns.Add("bookname");

dz.Columns.Add("companyname");

dz.Columns.Add("date");

dz.Columns.Add("kamot");

for (int j = 0; j < kr1.Rows.Count; j++)

{

for (int k = 0; k < kr1.Columns.Count; k++)

{

if (kr1.Rows[j][k].ToString().Contains(textBox4.Text))

{

dz.Rows.Add(kr1[j][0], kr1[j][1], kr1[j][2], kr1[j][3]);

k = kr1.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class lobtokef : Form

{

public lobtokef()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.kartis\_korehDataTable kr;

private void lobtokef\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.kartis\_koreh' table. You can move, or remove it, as needed.

this.kartis\_korehTableAdapter.Fill(this.liDataSet.kartis\_koreh);

kr = liDataSet.kartis\_koreh;

DataTable dz = new DataTable();

dz.Columns.Add("id ");

dz.Columns.Add("name ");

dz.Columns.Add("last name");

dz.Columns.Add("phone");

dz.Columns.Add("start day");

dz.Columns.Add("end day");

for (int j = 0; j < kr.Rows.Count; j++)

{

if (kr.Rows[j]["tokef"].ToString() == "לא")

{

dz.Rows.Add(kr[j][1], kr[j][2], kr[j][3]);

}

}

dataGridView1.DataSource = dz;

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class not\_availbe\_books : Form

{

public not\_availbe\_books()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.booksDataTable kr;

private void not\_availbe\_books\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.books' table. You can move, or remove it, as needed.

this.booksTableAdapter.Fill(this.liDataSet.books);

kr = liDataSet.books;

DataTable dd = new DataTable();

dd.Columns.Add("code");

dd.Columns.Add("name");

dd.Columns.Add("writername");

dd.Columns.Add("year");

dd.Columns.Add("language");

dd.Columns.Add("amount");

dd.Columns.Add("type");

dd.Columns.Add("age");

dd.Columns.Add("msomn");

dd.Columns.Add("raw");

dd.Columns.Add("company");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j]["amount"].ToString() == "0")

{

dd.Rows.Add(kr[j][0], kr[j][1], kr[j][2], kr[j][3], kr[j][4], kr[j][5], kr[j][6], kr[j][7], kr[j][8], kr[j][9], kr[j][10]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dd;

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class spakim : Form

{

public spakim()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.spakimDataTable kr;

private void spakim\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.spakim' table. You can move, or remove it, as needed.

this.spakimTableAdapter.Fill(this.liDataSet.spakim);

if(Form2.pass==true)

{

textBox1.Enabled = true;

textBox2.Enabled = true;

textBox3.Enabled = true;

textBox4.Enabled = true;

textBox5.Enabled = true;

textBox6.Enabled = true;

button1.Enabled = true;

button2.Enabled = true;

button3.Enabled = true;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void label2\_Click(object sender, EventArgs e)

{

}

private void label1\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox3.Text == "")

{

MessageBox.Show("חסר נתונים");

textBox3.Focus();

}

else

{

if (textBox2.Text == "")

{

MessageBox.Show("חסר נתונים");

textBox2.Focus();

}

else

{

if (textBox1.Text.Length != 10)

{

MessageBox.Show("נתון שגוי");

textBox1.Text = "";

textBox1.Focus();

}

else

{

if (textBox4.Text.Length != 10)

{

MessageBox.Show("נתון שגוי");

textBox4.Text = "";

textBox4.Focus();

}

else

{

if (find.IsValidEmail(textBox5.Text) == false)

{

MessageBox.Show("נתון שגוי");

textBox5.Text = "";

textBox5.Focus();

}

else

{

if (textBox6.Text == "")

{

MessageBox.Show("חסר נתונים");

textBox6.Focus();

}

else

{

DialogResult aa = MessageBox.Show(" האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "INSERT into spakim ([coname],[maname],[cophone],[manphone],[mail],[adress]) VALUES ('" + textBox3.Text + "','" + textBox2.Text + "','" + textBox1.Text + "','" + textBox4.Text + "','" + textBox5.Text + "','" + textBox6.Text + "')";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

spakim\_Load(sender, e);

}

}

}

}

}

}

}

}

private void button2\_Click(object sender, EventArgs e)

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "UPDATE [spakim] SET [maname]='" + textBox2.Text + "',[cophone]='" + textBox1.Text + "',[manphone]='" + textBox4.Text + "',[mail]='" + textBox5.Text + "',[adress]='" + textBox6.Text + "' where [coname]='" + textBox3.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

spakim\_Load(sender, e);

}

}

private void button3\_Click(object sender, EventArgs e)

{

DialogResult aa = MessageBox.Show("האם אתה בטוח ", "exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (aa == DialogResult.Yes)

{

OleDbCommand cmd = new OleDbCommand();

cmd.CommandText = "delete from spakim where [coname]='" + textBox3.Text + "'";

cmd.Connection = con;

con.Open();

cmd.ExecuteNonQuery();

con.Close();

spakim\_Load(sender, e);

}

}

private void textBox7\_TextChanged(object sender, EventArgs e)

{

kr = liDataSet.spakim;

DataTable dz = new DataTable();

dz.Columns.Add("coname");

dz.Columns.Add("maname");

dz.Columns.Add("cophone");

dz.Columns.Add("manphone");

dz.Columns.Add("mail");

dz.Columns.Add("adress");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j][k].ToString().Contains(textBox7.Text))

{

dz.Rows.Add(kr[j][0], kr[j][1], kr[j][2], kr[j][3], kr[j][4], kr[j][5]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

private void maskedTextBox1\_MaskInputRejected(object sender, MaskInputRejectedEventArgs e)

{

}

private void textBox1\_TextChanged\_1(object sender, EventArgs e)

{

if (textBox1.Text.Length > 0)

{

string s = textBox1.Text;

int i = s.Length - 1;

if (!char.IsNumber(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox1.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox1.Text = s1;

}

else

textBox1.Clear();

}

}

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

if (textBox2.Text.Length > 0)

{

string s = textBox2.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox2.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox2.Text = s1;

}

else

textBox2.Clear();

}

}

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

if (textBox3.Text.Length > 0)

{

string s = textBox3.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox3.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox3.Text = s1;

}

else

textBox3.Clear();

}

}

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

if (textBox4.Text.Length > 0)

{

string s = textBox4.Text;

int i = s.Length - 1;

if (!char.IsNumber(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox4.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox4.Text = s1;

}

else

textBox4.Clear();

}

}

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

if (textBox6.Text.Length > 0)

{

string s = textBox6.Text;

int i = s.Length - 1;

if (!char.IsLetter(s[i]))

{

MessageBox.Show("היקשתה נתון לא מתאים");

textBox6.Focus();

if (s.Length > 1)

{

string s1;

s1 = s.Substring(0, s.Length - 1);

textBox6.Text = s1;

}

else

textBox6.Clear();

}

}

}

//private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

//{

// i

//}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex != -1)

{

textBox3.Text = dataGridView1[0, e.RowIndex].Value.ToString();

textBox2.Text = dataGridView1[1, e.RowIndex].Value.ToString();

textBox1.Text = dataGridView1[2, e.RowIndex].Value.ToString();

textBox4.Text = dataGridView1[3, e.RowIndex].Value.ToString();

textBox5.Text = dataGridView1[4, e.RowIndex].Value.ToString();

textBox6.Text = dataGridView1[5, e.RowIndex].Value.ToString();

}

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.OleDb;

namespace liii

{

public partial class trash : Form

{

public trash()

{

InitializeComponent();

}

OleDbConnection con = new OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\li.mdb");

liDataSet.trashDataTable kr;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

kr = liDataSet.trash;

dataGridView1.Visible = true;

if (comboBox1.Text == "כרטיס קורא")

{

DataTable dz = new DataTable();

dz.Columns.Add("id ");

dz.Columns.Add("date");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j]["bitol"].ToString() == "k")

{

dz.Rows.Add(kr[j][1], kr[j][3]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

else if (comboBox1.Text=="השאלות")

{

DataTable dz = new DataTable();

dz.Columns.Add("id ");

dz.Columns.Add("book name ");

dz.Columns.Add("date");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j]["bitol"].ToString() == "h")

{

dz.Rows.Add(kr[j][1], kr[j][2], kr[j][3]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

else

{

DataTable dz = new DataTable();

dz.Columns.Add("id ");

dz.Columns.Add("book name ");

dz.Columns.Add("date");

for (int j = 0; j < kr.Rows.Count; j++)

{

for (int k = 0; k < kr.Columns.Count; k++)

{

if (kr.Rows[j]["bitol"].ToString() == "hz")

{

dz.Rows.Add(kr[j][1], kr[j][2], kr[j][3]);

k = kr.Columns.Count;

}

}

}

dataGridView1.DataSource = dz;

}

}

private void trash\_Load(object sender, EventArgs e)

{

// TODO: This line of code loads data into the 'liDataSet.trash' table. You can move, or remove it, as needed.

this.trashTableAdapter.Fill(this.liDataSet.trash);

}

}

}