משחק הזכרון

במשימה זו ניצור משחק זיכרון המשתמש בזוגות קלפים.

חוקי המשחק הידועים לנו במשחק הזיכרון תקפים גם במשחק זה :

במשחק יהיו זוגות של תמונות, ואנו נחשוף בכל פעם 2 קלפים. אם חשפנו זוג קלפים נקבל נקודה והקלפים יישארו חשופים, אחרת, נסתיר את הקלפים שחשפנו.

נחזור על התהליך עד...שייחשפו כל הזוגות הקיימים ונציג הודעת ניצחון מתאימה.

כל "קלף" יממומש ע"י תמונה (Image) אשר תוסתר, כשצריך, ע"י כפתור (Button).

לוח המשחק, בשלב זה, ימומש ע"י Grid בגודל 4X2 תאים.

מערכה

1

:

נפתח

פרוייקט

WpfApp

שלו

הראשי

ובחלון

חדש

)

MainWindow

(

ניצור

Grid

בגודל

8

)

4

X

2

תאים

(

.

תא

בכל

תמונה

נכניס

Image

.

כרצוננו

תמונות

.

שהכנסנו

של

זוגות

לוודא

יש

,

המשחק

חוקי

למען

)

לרווח

מומלץ

,

ולמען

היופי

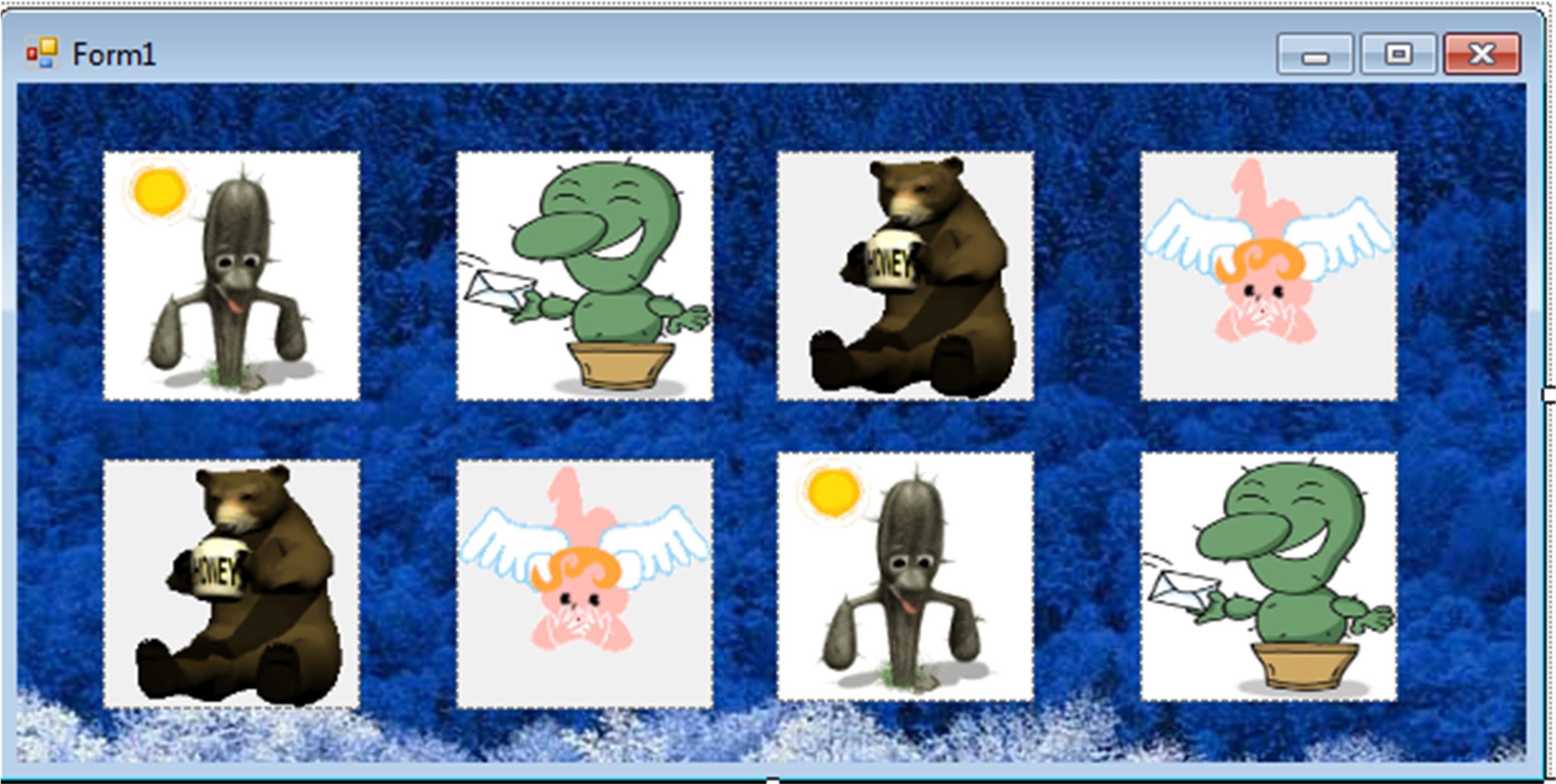
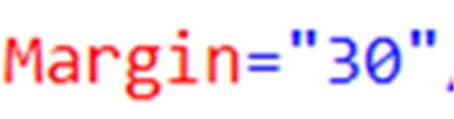
(

כל

בין

.

התמונות



למען הסדר הטוב, ניצור בפרוייקט תיקיית תמונות (Images) ובה נכניס את כל התמונות.



תיקייה

בפ

יצירת

עבור

:

רוייקט

ימני

על

,

הפרויקט

Add



NewFolder

לתיקייה

שנוצרה

שם

למשל

)

כרצוננו

ניתן

ואז

Images

.(



עבור

:

בתיקייה

,

לפרוייקט

תמונה

הוספת

ימני

התיקייה

,

על

Add



ExistingItem

נבחר

שנפתחת

בחלונית

...

ו

:

א

הקבצים

סוגי

כל

את

.

התיקייה

את

.

ב

התמונה

נמצא

בה

.

ג

,

עצמה

את

התמונה

.

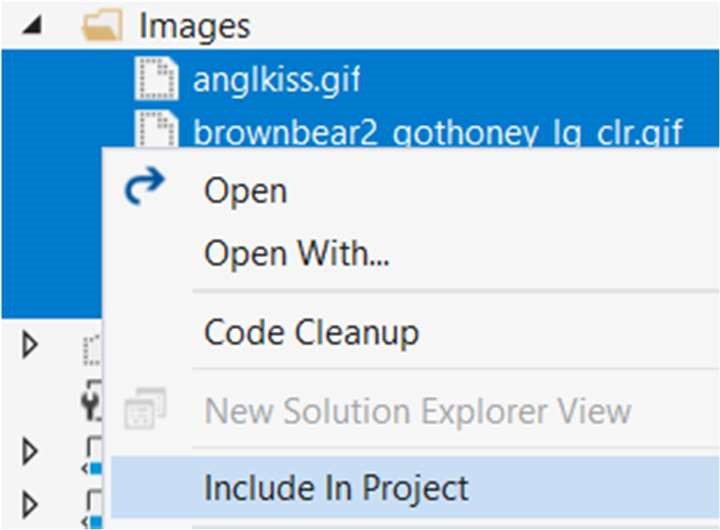
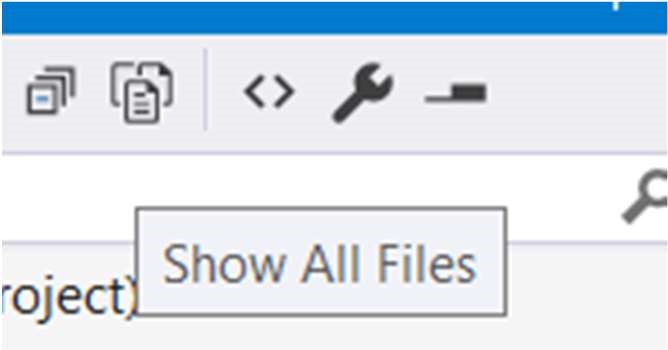
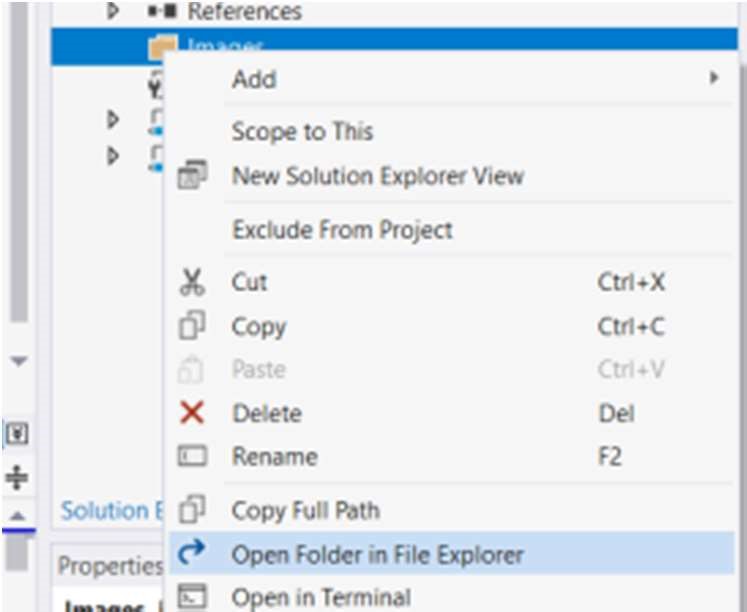
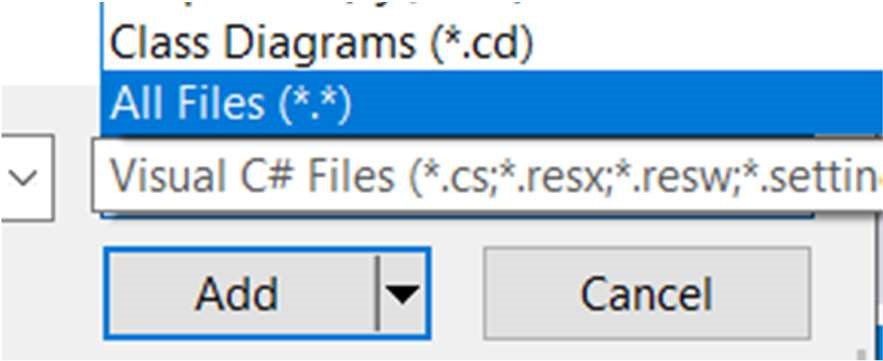
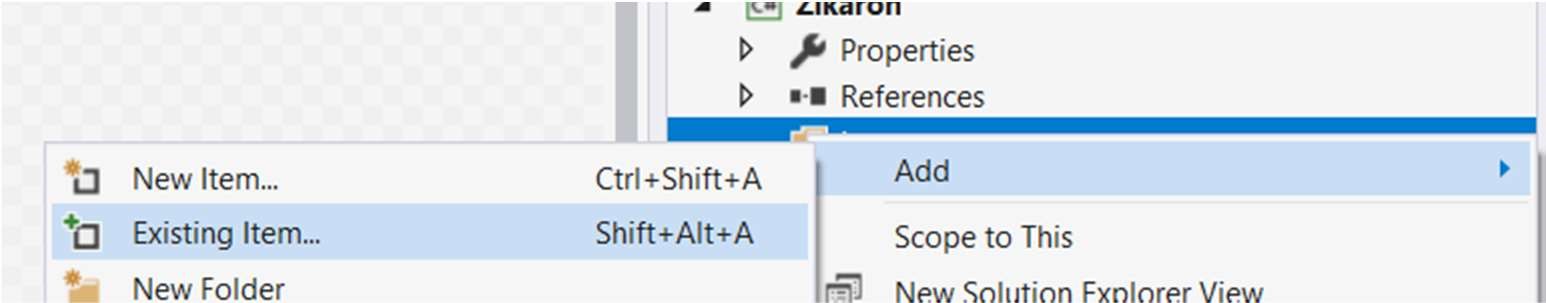
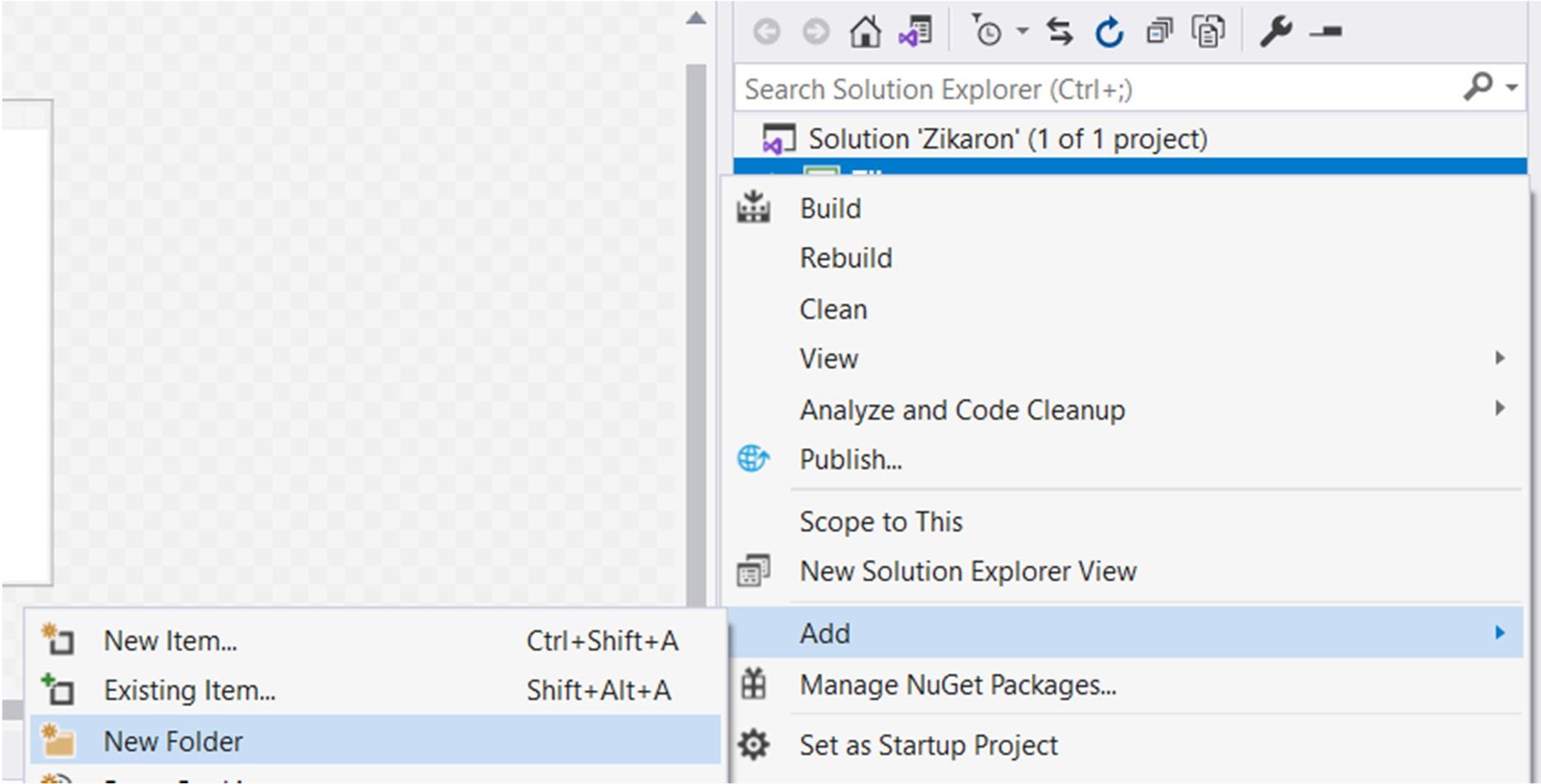
כפולה

לחיצה

י

"

ע



או...

ניתן לפתוח את תיקיית התמונות ע"י ימני ו- openFolederInFileExplorer .

ואז לגרור לתיקייה את התמונות שנרצה.

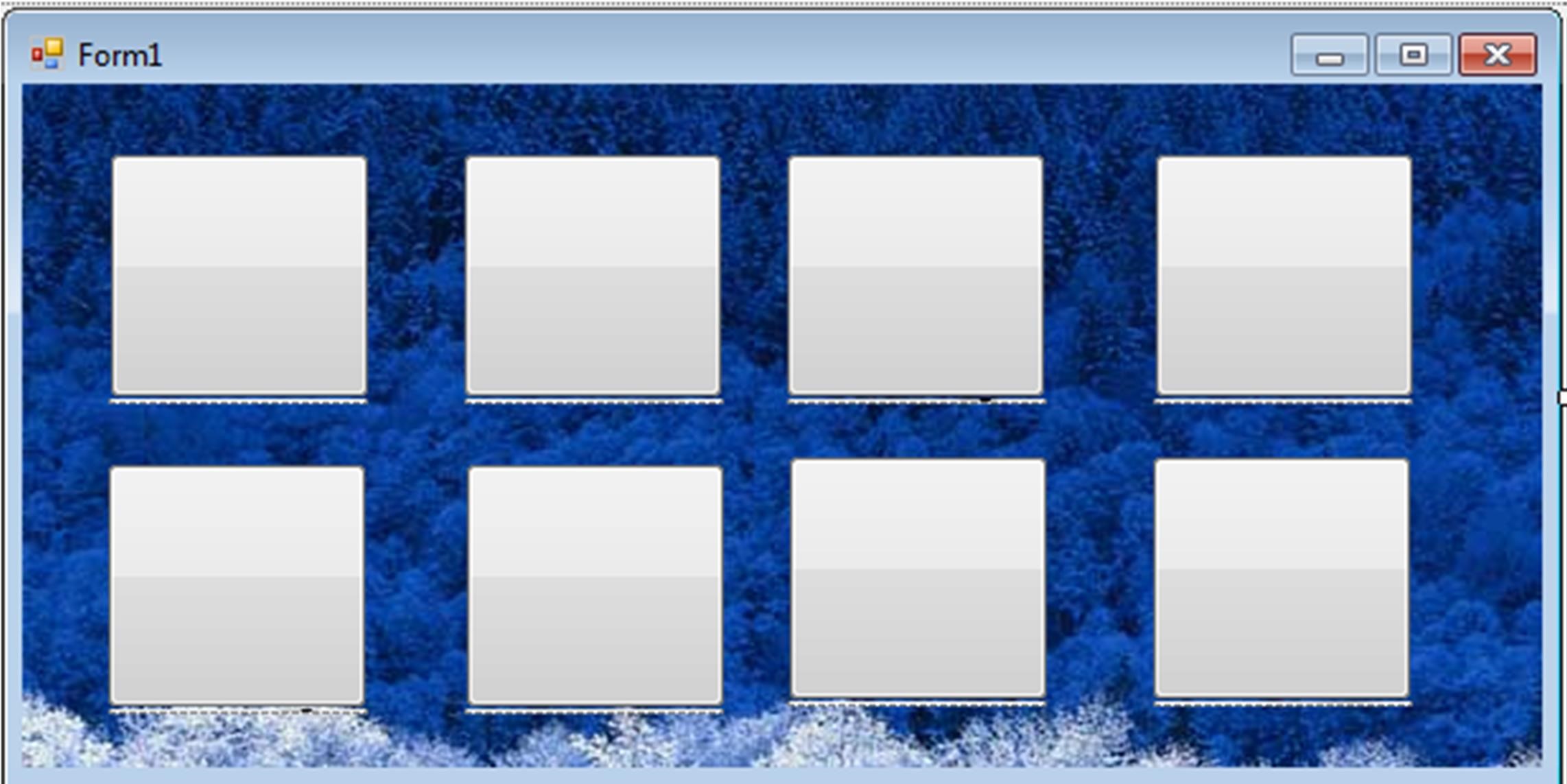
. showAllFiles את SolutionExplorer יש לבחור, בחלון ,VisualStudio -אח"כ , ב

ואז נראה בתיקיית התמונות( Images) את כל הקבצים של התמונות שגררנו, כשהם "שקופים".

נבחר את כל התמונות (או ימני עבור כל תמונה) ואז IncludeInProject יכניס את התמונות לפרוייקט, כך שנוכל להשתמש בהם.

ניתן גם לבחור עבור ה- Grid תמונת רקע – כך שיהיה יפה.

לשם כך נשתמש בחלונית מאפיינים של ה- Grid.



נכסה כל אחד מהתמונות בכפתור משלו.

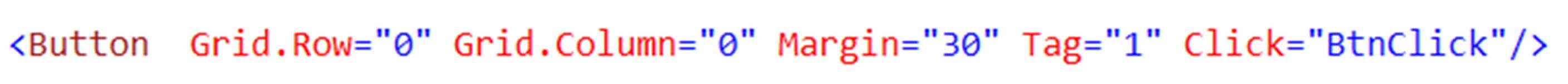
כדי לזהות כפתורים המכסים זוג תמונות ,ניתן לכל כפתור Tag המכיל את מספר התמונה.

כלומר שני כפתורים המסתירים את שני המופעים של תמונה מספר 1 ייקבלו שניהם Tag=1, ושני כפתורים המסתירים את שני המופעים של תמונה מספר 2 ייקבלו שניהם Tag=2, וכן הלאה.

נגדיר לכל הכפתורים פעולה אחידה עבור אירוע Click.

גם הכפתורים ייקבלו מרווח, כמו התמונות, כך שיכסו אותם בדיוק.

תזכורת: אם כפתור או תמונה נמצאים בתא של Grid ואין להם גודל מפורש – הם ייתפסו את כל גודל התא.



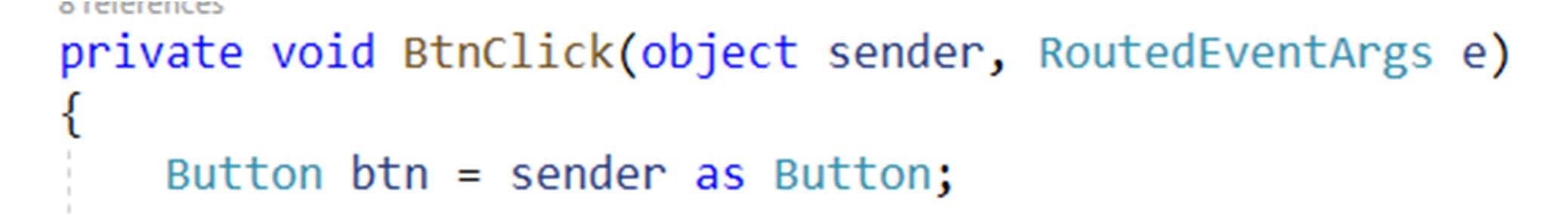
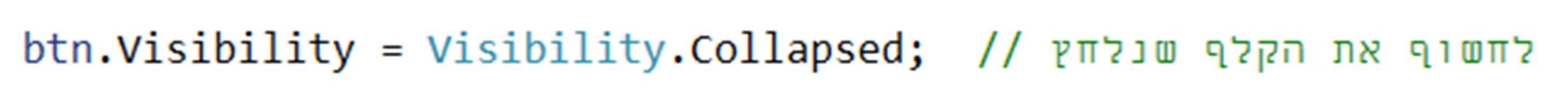
כל שנותר לנו עכשיו זה לכתוב את קוד המשחק, שיתבצע בעת ללחיצה על כפתור כלשהו.

מכיון שכל הכפתורים "מתנהגים" אותו הדבר נייצר פעולה אחת BtnClick , אשר תטפל באירוע הלחיצה, ונקשר אותה לכל הכפתורים הקיימים .

הפעולה מקבלת פרמטר sender שזה הרכיב( Component) שגרם לאירוע, במקרה שלנו זה יהיה הכפתור שגרם לאירוע.

נשמור אותו במשתנה מסוג Button .

כל לחיצה על כפתור – תגרום לנו להסתיר את הכפתור, כלומר לגלות את התמונה מתחתיו (את הקלף.)



אם חשפנו שתי תמונות שהם לא זוג, יש להסתיר אותם, אבל...

זה חייב להתבצע בלחיצה הבאה ,וזאת כדי לאפשר לשחקן לצפות בתמונה שנחשפה.

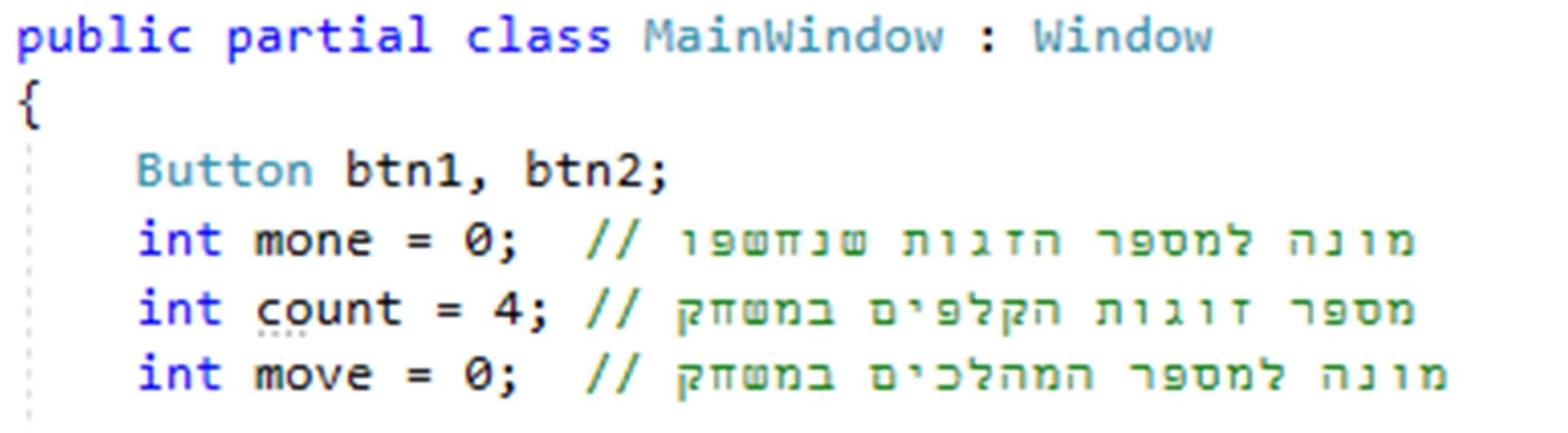
לכן.....אחרי לחיצה על כפתור ,נבדיל בין 3 מצבים אפשריים:

1. לחיצה על הכפתור הראשון בזוג.
2. לחיצה על הכפתור השני בזוג.
3. לחיצה על הכפתור הראשון בזוג הבא.

יש לשים  מתי מסתיים המשחק ואיך נגלה זאת בקוד.

נשמור את הכפתורים שנלחצו במשתנים btn1 ו- Btn2, בהתאמה, שניהם מסוג Button, כמובן.

למען התחרותיות במשחק ,ניתן להוסיף ספירה של מספר המהלכים ו...להציג זאת בהודעת הניצחון.



פתרון אפשרי עבור הקוד של הפעולה:



כל שנותר זה להריץ, ולבדוק שהכל עובד, לשביעות רצוננו

