

***משחקי קלפים***

## רקע-

משחקי קלפים הם כל אותם המשחקים שבהם הקלפים משמשים ככלי המרכזי במשחק(למשל מונופול לא נחשב משחק קלפים כי הקלפים הם רק תוסף למשחק ולא החלק האינטגרלי בו).  
משחקים קלפים רבים קיימים אך רק מעטים מהם שמשוחקים עם חפיסה תקנית בעלי חוקים פורמליים.  
מרבית משחקי קלפים מנצלים את העובדה שניתן לזהות קלפים רק מצד אחד, כך שרק השחקן שמחזיק את הקלפים יכול לדעת מה יש לו, אבל לא מה יש לאחרים. זאת אחת הסיבות לכך שבד"כ משחקי קלפים מאופיינים עם מזל.   
ישנם כמה סוגים של משחקי קלפים:

* *Trick taking games*- בהם השחקן צריך לצבור כמה שיותר קלפים (או כמה שפחות קלפים רעים).  
  למשל המשחק ברידג'.
* *משחקי התאמות*- לנסות להשיג כמה שיותר סטים של קלפים מאותו הסוג, כל סט מקדם אותך לניצחון. דוגמא למשחק: רמי
* *Shedding games*- משחקים בהם המטרה להיפטר מכמה שיותר קלפים ביד. דוגמא למשחקים: ספיד, רמי-קוב.
* *משחקי השוואה-* המטרה להגיע למצב שמספר הקלפים ביד של השחקן זהה למספר מוגדר מראש למשל: 21(בלאק-ג'ק).
* *משחקי מזל*- משחקי קזינו שבהם המנצח זוכה בכסף (או משהו יקר ערך אחר), למשל פוקר.
* *משחקי סוליטר(או סבלנות)-* משחקי המשוחקים בדר"כ ע"י אדם אחד. השחקן מתחיל ממבנה אחד ומטרתו היא להיפטר מהמבנה או לבנות להיפטר מהחפיסה בידו ע"י הצבה של קלפים במבנה.

כמובן שכל אחת מהקטגוריות יכולה להיות משולבת עם קטגוריה אחרת.

בפרק נבנה משחק סוליטר.   
בסוף פרק זה יהיה לנו, לא רק משחק קלפים לטלפון, אלא גם אחלה תבנית למשחקי קלפים אחרים שנרצה לבנות בעתיד.  
נראה כמה טכניקות חדשות, הכוללות: שימוש בקונפיגורציות XML ועיצוב משחקים לטלפונים סלולריים.

## Set up-

במהלך השיעור נשתמש ב-asset שזמין באופן ציבורי: *Vectorized Playing Cards* של Chris Aguliar.  
את ה-asset ניתן להוריד [כאן](https://github.com/maoz-grossman/Card_Game/tree/master/Vectorized%20Playing%20Cards), או באתר <http://book.prototools.net/?page_id=519> למטה, היכן שכתוב Starter Package מתחת מופיע ה- asset: "C32\_Prospector\_Starter.unitypackage".   
ניצור פרויקט חדש ב-unity (אין חשיבות אם D2 או D3). ניתן לפרויקט שם, נייבא את ה-asset שהורדנו לפרויקט ע"י גרירה.  
בשיעור נשתמש בסצנה שקיבלנו עם ה-asset שהורדנו: \_Prospector\_Scene\_0 .  
ניכנס לאובייקט המצלמה, ונוודא כי באינספקטור שלה היא מוגדרת כך: Position(0,0,-4),Rotation(0,0,0),Scale(1,1,1)   
projection: Orthographic ו-size:10, ונוודא שיחס המסך הוא 4:3 (מעל חלון המשחק).

# Build Setting-

את המשחק נבנה לטלפון, אך כמובן שניתן לבנות אותו לאיזו פלטפורמה שתעדיפו ובלבד שתתאימו אותו לאותה הפלטפורמה.  
תזכורת: אם נרצה לשנות את פלטפורמת המשחק שלנו:1. נלחץ פעמיים על הסצנה (\_Prospector\_Scene\_0) כדי לפתוח אותה.   
2.מהתפריט הראשי נבחר *file-> build setting* שיפתח לנו את החלון הבא:

3. גררו את הסצנה לחלון scene In Build, או שלחצו על *Add Open Scene* להוספת הסצנה (אני מזכיר שהסצנה שנעבוד עליה היא \_Prospector\_Scene\_0).   
4. בחרו בפלטפורמה המתאימה (אנחנו נעבוד עם אנדרואיד ,אבל שוב למי שיש מכשיר של apple מוזמן לשנות לפלטפורמה שמתאימה לו), ואח"כ *Switch Platform.* Unity מייבא כעת את כל התמונות כדי להתאים אותם לברירת המחדל של הפלטפורמה שבחרנו. ונסגור את החלון.  
(אין צורך ללחוץ *Build* עדיין, אנחנו נעשה את זה רק בסוף בניית המשחק).

## ייבוא תמונות כ-Sprites:

נצטרך לייבא את התמונות כראוי כדי להתאימן לשימוש כ-Sprites. תזכורת: Sprite היא תמונת דו ממד שניתן לבצע עליה מניפולציות כמו הגדלה, הזזה, סיבוב וכדו'.  
  
1. נפתח את תיקיית \_*Sprties* בחלון הפרויקט, ונבחר את כל התמונות שבו- או ע"י בחירה באחד ואז בליחצה על ctrl + חץ למטה נסמן את כל התמונות עד שנגיע לתחתית התיקייה, או שנבחר את התמונה הראשונה ואז shift + לחיצה על התמונה האחרונה בתיקייה.  
  
2. ניכנס לאינספקטור (אם סימנו את כל התמונות האינספקטור משפיע על כולן). ונשנה את ה-*Texture type* שלהן ל-*Sprite(2D and UI)*. השינוי אמור להשפיע על כלל התמונות שסימנו.   
אם נסתכל בחלון הפרויקט ליד כל תמונה מופיע משולש קטן, אם נלחץ נראה sprite שבתוך התמונה שלנו עם אותו השם.  
   
3. בחר את תמונה *Letters* בחלון הפרויקט. כל התמונות שייבאנו ה-*Sprite Mode* אמור להיות מוגדר כ-*Signle* וזה מתאים כי כל תמונה מייצגת רק sprite אחד . לעומת זאת התומנה *Letter* אמורה לייצג יותר מספרייט אחד (כל אות בו אמורה להיות ספרייט בפני עצמה). לכן נצטרך לשנות את ההגדרות שלה כך: בחלון האינספקטור נשנה את ה-*Sprite Mode* של  *Letter* ל-*Multiple* ונלחץ apply כדי ליישם את השינוי.  
 נבחר ב-*Sprite Editor,* וחלק את התמונה בהתאם לגודל קבוע עבור כל אות. להסבר מורחב ניתן למצוא בסיכום למשחק חלליות ,פרק אנימציות תת-פרק אפקט פיצוץ.  
ונשמור את הסצנה בינתיים.

## בניית קלף מ-sprite

במהלך הפרויקט אנחנו הולכים לבנות חפיסה שלימה של קלפים מ-21 הקלפים שייבאנו. זה מאפשר לחסוך במקום בבנייה האחרונה של המשחק.  
ספויילר: המבנה של קלף כפי שנראה בהמשך יהיה בצורה של כמה אובייקטים מהצורה כשלהי (כגודל המספר אותו הוא מייצג + שני צורות בצדדים ליד המספר, לא כולל הקלפים המיוחדים: ממשפחת המלוכה או ג'וקרים):

# שימוש בתבניתXML בקוד-

ראשית ניצור שלושה סקריפטי- c#בתיקייה \_Script עם השמות *Card ,Deck* ו- *Prospector*.

* **Card-** מחלקה לקלף אינדיבידואלי בחפיסה. המחלקה מכילה גם מחלקה פנימית CardDefinition (שמכילה את כל המידע על מיקום הספרייטים בקלף לכל מספר קלף). ומחלקת Decorator (שמחזיקה מידע לקובץ xml).
* **Deck-** החפיסה, מפרש את המידע שב- DeckXML.xml ומשתמש במידע הזה כדי ליצור חפיסה.
* **Prospector-** המחלקה מנהלת את כל המשחק במבט על. כשנוצרת חפיסה מהמחלקה deck המחלקה prospector הופכת אותם לקלפי משחק. המחלקה אוספת את הקלפים לכמה ערימות (למשל ערימה שממנה מושכים או ערימה שזורקים אליה קלפים משומשים) ומנהלת את לוגיקת המשחק.