

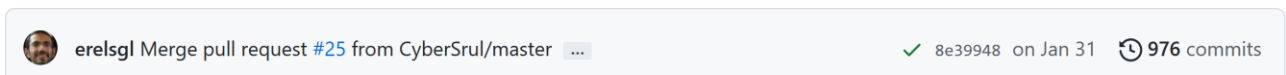
מטלה שבועית – מבוא ליוניטי

השאלות במטלה זו נועדו לעזור לכם להרחיב את הידע שלכם בנושאים שנלמדו בהרצאה.

- יש להעלות את קוד-המקור לגיטהאב של ה"אירגון" (organization) שפתחתם עבור הצוות. היעזרו בקובץ ([4-github-and-unity](#)) שנמצא בתיקיה של השבוע הנוכחי בקורס, כדי לעלות את הפרויקט שבניתם בצורה יעילה ובלי קבצים מיותרים.
- יש לוודא שכל חברי הצוות יכולים לעבוד עם המאגר בגיטהאב – להוריד אותו, לבצע שינויים, ולהגיש (בהיסטוריה של המאגר בגיטהאב, נבדוק שיש הגשות של כל חברי הצוות).
- יש להעלות את המשחקים עצמם לחשבון של הצוות, שתפתחו באתר [itch.io](#).
- יש לשים קישור מקובץ ה-[Readme.md](#) בגיטהאב למשחק ב [itch.io](#), וכן קישור מהערות המשחק ב [itch.io](#) לגיטהאב של המשחק, כך שמי שמוצא את המשחק שלכם באחד האתרים יוכל להגיע לשני.
- יש להגיש בתיבת ההגשה במודל שני קישורים: קישור לגיטהאב וקישור ל-[itch.io](#).

דגשים:

- יש לכתוב קוד קריא, מתועד, בלי מספרי קסם וכו' – כמו שלמדתם בקורסים הקודמים.
- יש להשתמש בפעולות קלט `InputAction` כמו שלמדנו בשיעור (ולא במערכת הקלט הישנה).
- יש לשים במאגר-גיטהאב שלכם, בתיקיה [github/workflows](#). (שימו לב לנקודה בהתחלה), את הקובץ `linter.yml` שנמצא באתר הקורס. אין לשנות את הקובץ. אם הבדיקות עברו בהצלחה, תוכלו לראות סימן וי ירוק ליד הקומיט שלכם בגיטהאב, כמו שרואים כאן:



פרטים נוספים ניתן לראות בסרטון: <https://www.youtube.com/watch?v=BCrtoZ04L1Y>

א. רכיבים

- בהרצאה בנינו רכיב בשם **Mover** המזיז עצם-משחק בכיוון קבוע ובמהירות קבועה. בנו רכיב חדש בשם **Oscillator**, המזיז עצם-משחק בתנועת מטוטלת, למשל: ימינה עד גבול מסוים, ואז שמאלה עד גבול מסוים, וחוזר חלילה.
- מרכז התנודה צריך להיות המקום הנוכחי של העצם בסצנה (למשל אם אני שם אותו במיקום 3,2, אז העצם יזוז ימינה ושמאלה סביב נקודה זו).
- מהירות העצם צריכה להשתנות בהדרגה, כמו במציאות (העצם נע ימינה, מאט בהדרגה, נעצר, נע שמאלה, מאיץ בהדרגה, מאט בהדרגה, נעצר וכו'). רמז: השתמשו בפונקציה טריגונומטרית.
- הגדירו את הרכיב באופן כמה שיותר כללי – עם כמה שיותר משתנים המאפשרים לשלוט בכיוון התנודה וגודלה מתוך יוניטי.

- הדגימו את הרכיב שלכם על כמה עצמים, עם ערכים שונים לפרמטרים.
- 2. בנו רכיב בשם **Rotator**, המסובב עצם-משחק במהירות נתונה סביב עצמו (באיזה ציר-סיבוב צריך להשתמש?)
- 3. בנו עצם-משחק בצורת כדור, הגדל בהדרגה וקטן בהדרגה, כמו לב פועם.
- 4. בנו רכיב המאפשר לשחקן להסתיר את העצם שלו (העצם שהרכיב שייך אליו) בלחיצת כפתור, ולהציג אותו מחדש בלחיצת כפתור. רמז: השתמשו ברכיב האחראי על הצגה של העצם. השתמשו ב-InputAction כפי שנלמד בהרצאה.

הגשה: כדי להדגים את כל הרכיבים בסעיף זה, אפשר ליצור פרויקט אחד עם סצנה אחת, ולשים בה עצמים שונים המדגימים את הרכיבים השונים.

- העלו את הפרויקט שיצרתם לגיטהאב, וכן לאתר itch.io.
- מומלץ לכתוב ליד כל עצם מה בדיוק הוא מדגים. לשם כך אפשר להוסיף טקסט לסצנה באופן הבא:
- הוסיפו עצם-משחק חדש ריק (**Create Empty**)
- הוסיפו לו רכיב חדש (**Add Component**) מסוג **Text Mesh Pro – Text**. (*)
- שנו את מאפייני הרכיב כדי לקבוע את הגודל, הצבע, הכיוון והשפה. אפשר לכתוב בעברית או אנגלית.
- (*) אם אתם לא מוצאים את **Text Mesh Pro**, ייתכן שאתם צריכים להתקין אותו באופן חד-פעמי. לשם כך היכנסו ל **Window** ואז ל **Package Manager**, חפשו את החבילה **Text Mesh Pro** והתקינו אותה.

ב. מצלמות ונקודות-מבט

1. קיראו כאן: <https://docs.unity3d.com/Manual/class-Camera.html> על מצלמות ביוניטי.
 2. בנו "משחק" שבו השחקן יכול לזוז בעולם דו-ממדי בעזרת מקשי החיצים, ובמקביל הוא רואה את המיקום שלו על מפה קטנה (mini-map) בצד ימין למעלה.
 3. בטאב "Simulator" של עורך יוניטי, אפשר ללחוץ על הכפתור **Rotate** כדי לסובב את הסימולטור, וכך לדמות מצב שהשחקן מסובב את הטלפון שלו ממצב אופקי לאנכי ולהיפך. שימו לב, שכאשר אתם מסובבים את הסימולטור למצב אנכי, כל העצמים בסצנה נראים גדולים יותר.
- א. הסבירו מדוע (רמז: זה קשור לתכונה **Size** של המצלמה).
 - ב. כתבו רכיב שיפתור את הבעיה, ויוודא שכל העצמים ייראו באותו גודל כשמסובבים את הטלפון (רמז: כתבו סקריפט שמשנה את השדה **orthographicSize** של המצלמה. אין לשנות את שאר העצמים במשחק).

הגשה:

- בכל השאלות אין צורך לבנות משחק מלא, אלא רק "משחק" מינימלי המדגים את פעולת המצלמה.
- בשאלה זו יהיה לכם נוח יותר לכתוב פרויקט נפרד עבור כל סעיף. אבל אפשר גם לכתוב משחק אחד המשלב את שני הסעיפים.
- העלו לגיטהאב ולאתר itch.io את הפרויקט/ים שכתבתם.