***Twine***

תוכן עניינים

[מה זה Twine ? 3](#_Toc40348770)

[הורדה 3](#_Toc40348771)

[יצירת סיפור חדש 4](#_Toc40348772)

[פורמט 5](#_Toc40348773)

[פיתוח הסיפור 6](#_Toc40348774)

[שמירת הקבצים 6](#_Toc40348775)

[יצירת עמודים 7](#_Toc40348776)

[תוכן העמוד 8](#_Toc40348777)

[הוספת תגיות 9](#_Toc40348778)

[תצוגה 9](#_Toc40348779)

[Debug and Play 10](#_Toc40348780)

[Debug 11](#_Toc40348781)

[Play 12](#_Toc40348782)

[שפות תכנות 12](#_Toc40348783)

[HTML 12](#_Toc40348784)

[JavaScript 13](#_Toc40348785)

[הוספת מוזיקה לעמוד המשחק 13](#_Toc40348786)

[ערכית תוכן העמוד 15](#_Toc40348787)

[עריכת טקסט בסיסית 15](#_Toc40348788)

[הוספת קישורים 15](#_Toc40348789)

[Edit Story Stylesheet 17](#_Toc40348790)

[יישור טקסט וכתיבת טקסט בעברית 18](#_Toc40348791)

[שינויים לעמודים ספציפיים 19](#_Toc40348792)

[רווח בין אותיות 19](#_Toc40348793)

[שינוי צבע הטקסט 20](#_Toc40348794)

[שינוי צבע הלינק 20](#_Toc40348795)

[גופן הטקסט 22](#_Toc40348796)

[שינוי גודל הטקסט 23](#_Toc40348797)

[הוספת תמונות 23](#_Toc40348798)

[הגדרת תמונת רקע 24](#_Toc40348799)

[הוספת כפתורי שמירה וטעינה 25](#_Toc40348800)

[משתנים 26](#_Toc40348801)

[הפקודה Set: 27](#_Toc40348802)

[הפקודה Print: 27](#_Toc40348803)

[תנאים 27](#_Toc40348804)

[לולאות 28](#_Toc40348805)

[העלמת חץ החזרה אחורה 28](#_Toc40348806)

[ביבליוגרפיה 29](#_Toc40348807)

# מה זה Twine ?

תוכנה לבניית משחק טקסט מתקדם. Twine הוא כלי בקוד פתוח לספר סיפורים אינטראקטיביים.

אינך צריך לכתוב קוד כלשהו כדי ליצור סיפור פשוט עם Twine אך אתה יכול להרחיב את הסיפורים שלך עם משתנים, היגיון מותנה, תמונות, CSS ו- JavaScript כשתהיה מוכן.

Twine מפרסם ישירות ב- HTML כך שתוכל לפרסם את עבודותיך כמעט בכל מקום. כל דבר שתיצור איתו הוא בחינם לחלוטין ואתה יכול להשתמש בו בכל דרך שתחפוץ, כולל למטרות מסחריות.

נוצר על ידי כריס קלימאס בשנת 2009 וכעת הוא מתחזק על ידי חבורה שלמה של אנשים במאגרים שונים.

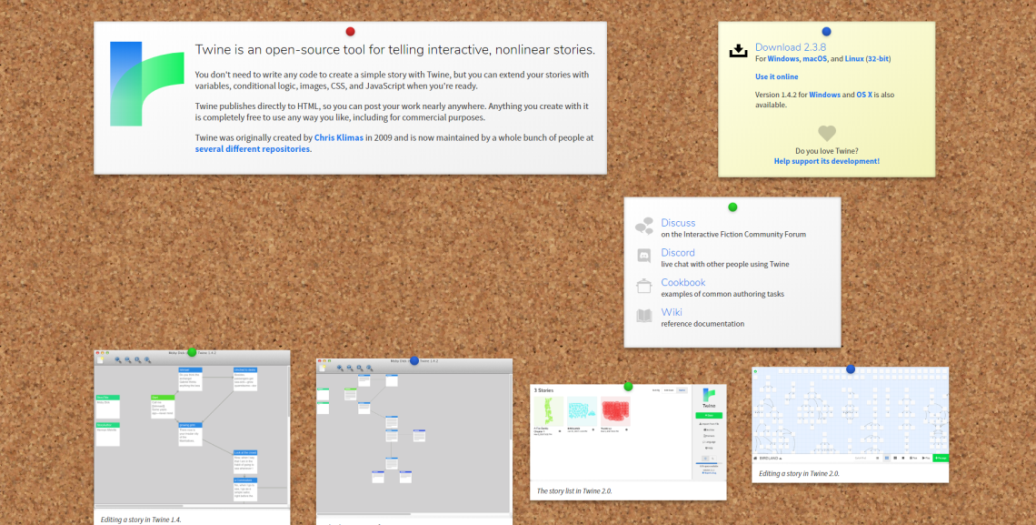
# הורדה

ניתן להוריד Twine דרך האתר הרשמי <http://twinery.org/>, אנו נשתמש ב Twine 2.

דרך נוספת **המומלצת** לשימוש שאינה מצריכה הורדה בכלל כוללת שימוש אינטרנטי בתוכנה (ניתן להשתמש בה דרך הכניסה לאתר הרשמי או אפילו ניתן לחפש בגוגל). שימוש בתוכנה דרך האינטרנט זה אומר שאתה לא צריך ליצור חשבון כדי להשתמש ב- Twine, וכל מה שאתה יוצר לא נשמר בשרת במקום אחר - הוא נשאר ממש בדפדפן שלך.

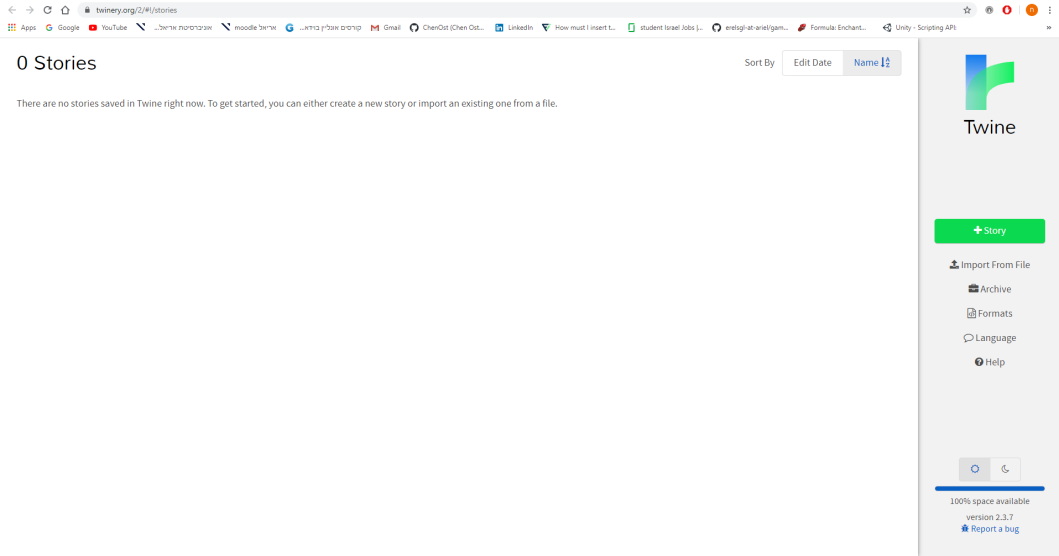
לכן ישנם שני דברים חשובים שיש לזכור:

1. מכיוון שעבודתך נשמרת רק בדפדפן שלך, אם תנקה את הנתונים שנשמרו ב cookies, תאבד את עבודתך! באפשרותך לשמור סיפורים בודדים לקבצים באמצעות התפריט הנמצא בכל סיפור. תמיד ניתן לייבא מחדש את קבצי הארכיון וגם את קבצי הסיפור.
2. כל מי שיכול להשתמש בדפדפן זה יכול לראות ולעשות שינויים בעבודה שלך. אז אם יש לך אח ילד חטטני, בדוק כיצד להקים לעצמך פרופיל נפרד.



# יצירת סיפור חדש

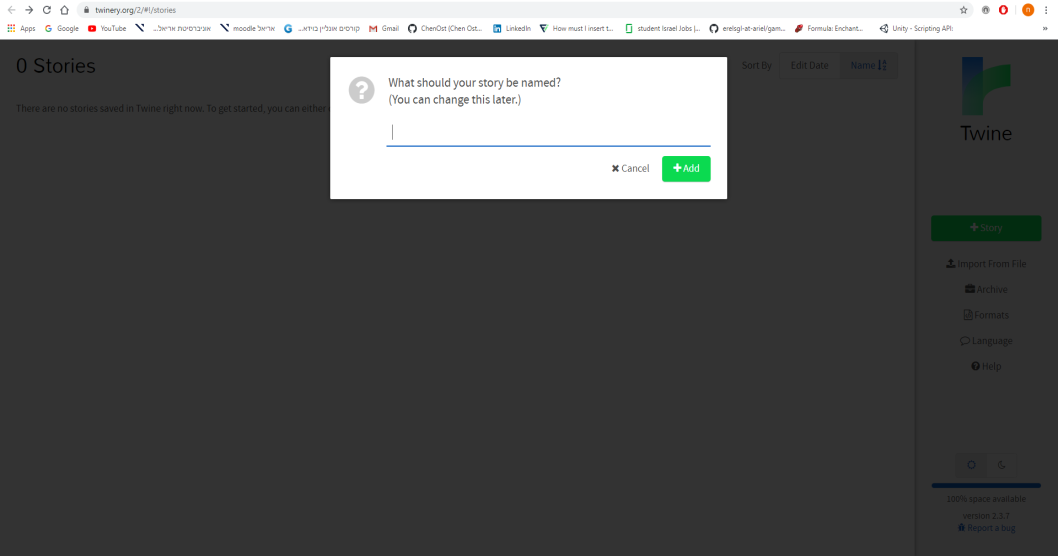
על מנת ליצור סיפור חדש ב Twine יש לעשות את השלבים הבאים:

**בשלב הראשון:** נלחץ על **+ Story** לאחר מכן יפתח לנו חלון חדש.

**(1)**

**בשלב השני:** יש לבחור שם לסיפור, אל חשש ! ניתן לשנות את השם גם בשלב יותר מאוחר.

**בשלב השלישי:** לאחר שבחרנו שם שאנו מרוצים ממנו נלחץ על **+ Add**.



**(2)**

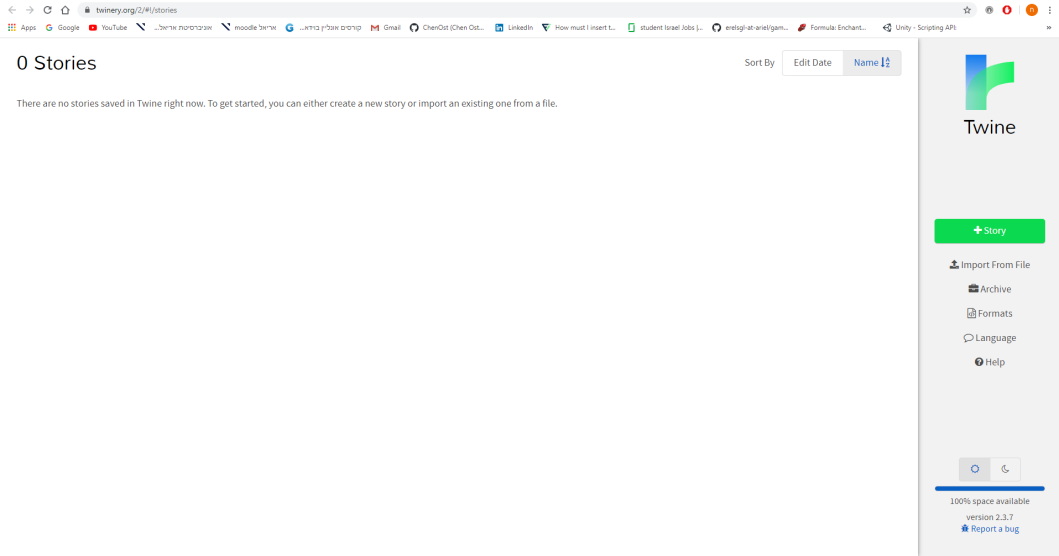
**(3)**

לאחר שהשלמנו את כל השלבים, נוכל לראות את הסיפור החדש שיצרנו במסך הראשי. ככל שהסיפור יהיה מפותח יותר, כך התמונה תשתנה בהתאם מעיגול אחד למספר רב של עיגולים, מספר העיגולים נקבע לפי כמות העמודים שנמצאים בסיפור ובהתאם לצורה שבה הם מסודרים בתוך הסיפור.

# פורמט

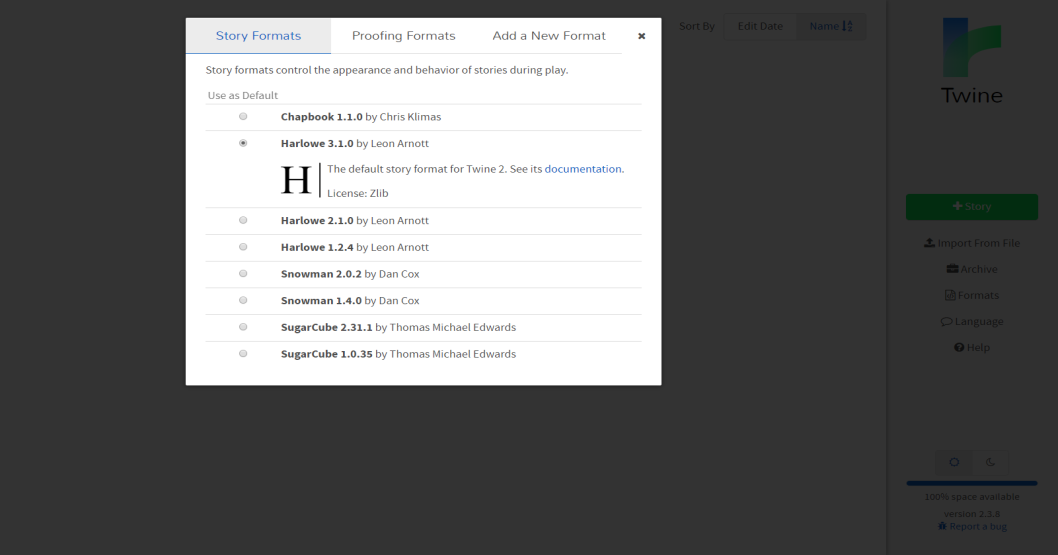
לפני שאנו מתחילים ביצירת הסיפור החדש שלנו, יש לוודא שהפורמט שבו נכתוב את הסיפור הוא העדכני ביותר. הפורמט מכיל את כל הפונקציות, קוד והמשתנים שבעזרתם נוכל לעצב ולבנות את הסיפור בצורה הנוחה והטובה ביותר. על מנת לבדוק באיזה פורמט אנו משתמשים, נעבור על השלבים הבאים:

**בשלב הראשון:** נלחץ על האפשרות **Formats.**

****

**(1)**

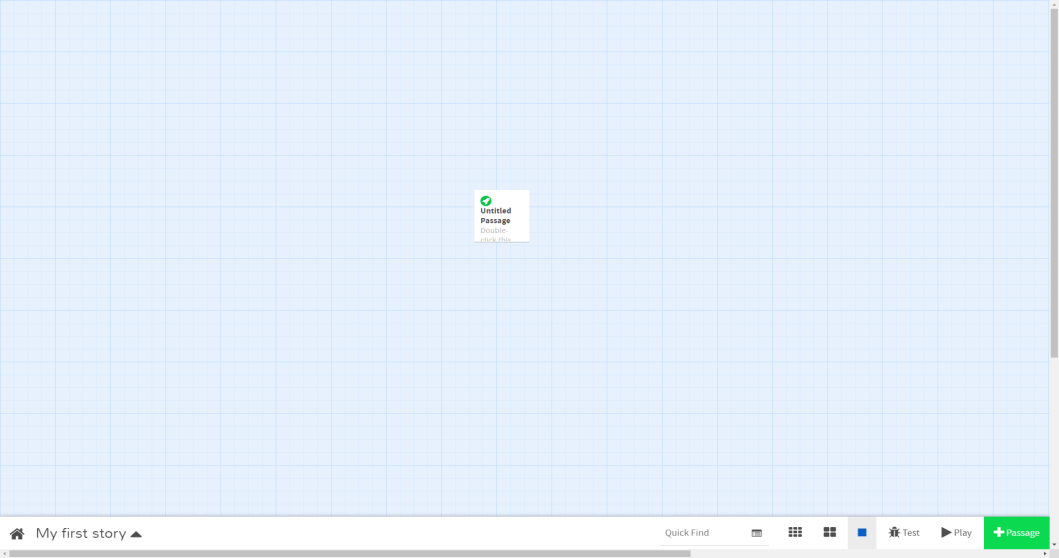
**בשלב השני:** יפתח לנו חלון קטן עם רשימת הפורמטים שאנו יכולים להשתמש בהם. יש לוודא שהגרסה המתקדמת ביותר היא המסומנת. ניתן לראות שהיא מסומנת בעזרת העיגול המלא. דבר נוסף שניתן להבחין בו הוא הקישור documentation, הקישור יכיל את כל ההסברים שאנו צריכים לגבי הפונקציות, המשתנים והשימושים האחרים שניתן לבצע בפורמט זה.



**(2)**

# פיתוח הסיפור

כשאנו נכנסים לסיפור שלנו, יופיע המסך הבא:



במסך זה נוכל לפתח את הסיפור שלנו ולהוסיף לו רבדים שונים, אך לפני שנסביר על כל האפשרויות השונות שהמסך הזה מכיל קודם נראה כיצד עלינו לשמור נכון את הסיפור שלנו על מנת למנוע עוגמת נפש בעתיד. כפי שכבר ציינו מקודם, חשוב מאוד לשמור את הקבצים על מנת שלא הכל ימחק לנו ולא תהיה דרך לשחזר אותם!.

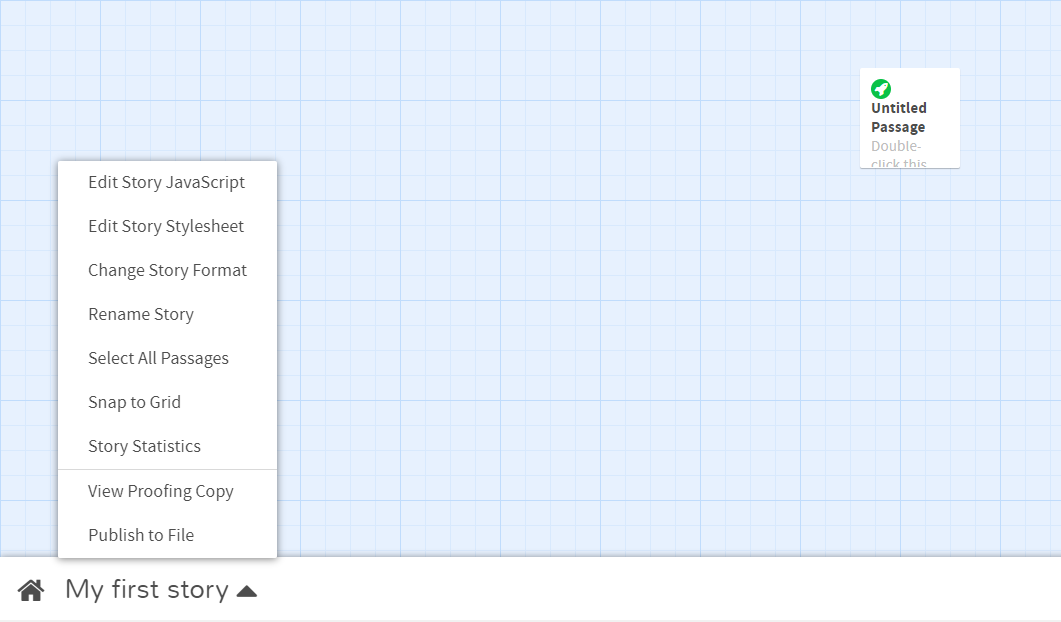
## שמירת הקבצים

על מנת לשמור את הקבצים יש לעבור על השלבים הבאים:

**בשלב הראשון:** נבחר תיקייה ייעודית במחשב שלנו, שתכיל את כל הקבצים של הסיפורים השונים.

**בשלב השני:** נלחץ על החץ שליד השם שבחרנו לסיפור שלנו, נמצא בפינה שמאלית למטה. החץ יפתח לנו מספר אפשרויות שונות.

**בשלב השלישי:** לאחר מכן נבחר באפשרות Publish to file, נחפש את התיקייה שבחרנו בסעיף הראשון ובה נשמור את הקובץ html שנוצר. **חשוב לשים לב!** במידה ואתה לוחץ על **CTRL + S** והתוכנה מציע לך לשמור את הקבצים בתיקייה הייעודית שבחרתהקובץ לא בהכרח ישמר!!!



**(2)**

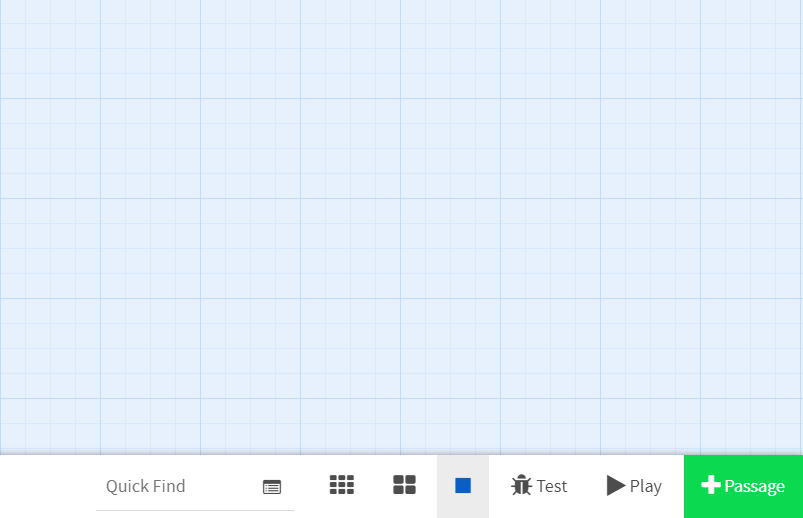
**(3)**

## יצירת עמודים

באמצע המסך מופיע לנו ריבוע קטן שניתן להזיז. הריבוע הזה מייצג עמוד בתוך הסיפור שלנו במידה ואנו יוצרים משחק סיפורי, בנוסף הוא יכול לייצג שיחה בין דמויות. ניתן להוסיף עמודים חדשים בכמה דרכים שונות:

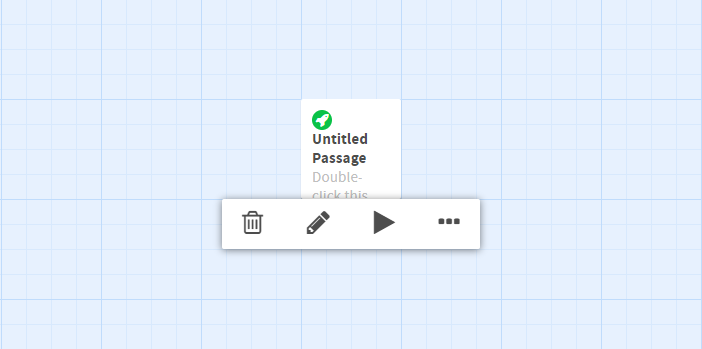
**דרך ראשונה:** בעזרת לחיצה על **+ Passage** הנמצא בפינה הימנית התחתונה של המסך מוסיף ישירות לסיפור שלנו עמוד נוסף.

**דרך שנייה:** את הדרך הזו ניתן לראות בהמשך [כאן](#_הוספת_קישורים) ניתן להוסיף עמוד חדש כאשר אנו יוצרים קישור מעמוד קיים לעמוד חדש.



**לחיצה כאן תוסיף לסיפור שלנו עמוד נוסף**

כאשר נעביר על העמוד את העכבר או אם נלחץ עליו יופיעו לנו 4 אפשרויות:



**(1)**

**(4)**

**(3)**

**(2)**

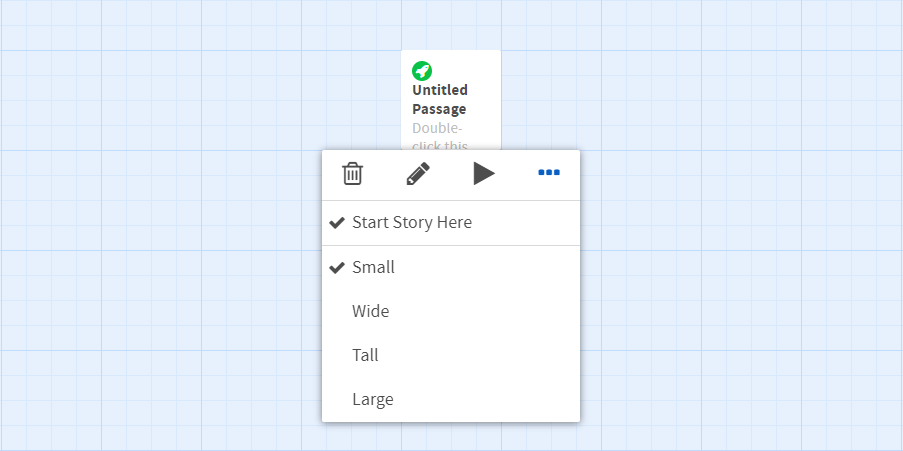
**אפשרות 1:** מחיקת העמוד.

**אפשרות 2:** עריכת העמוד (עוד מידע על עריכת העמוד ניתן למצוא כאן).

**אפשרות 3:** התחלת ה- Debug מהעמוד הנוכחי (עוד מידע על Debug ניתן למצוא [כאן](#_Debug)).

**אפשרות 4:** אפשרות זו מתחלקת למספר אפשרויות נוספות:

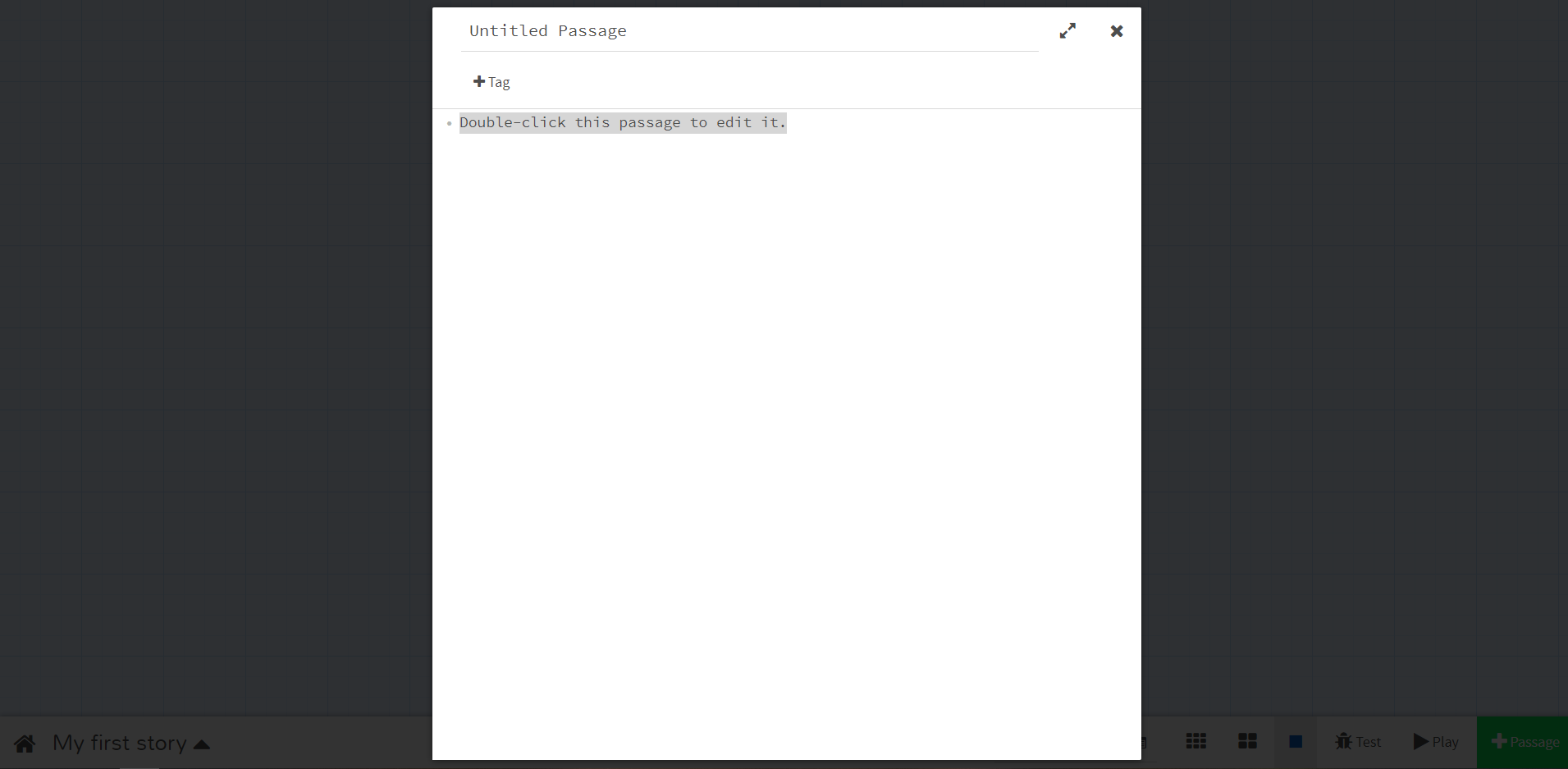
* **Start Story Here** – מאפשר להפוך את העמוד לנקודת ההתחלה של הסיפור, העמוד יסומן כנקודת התחלה של הסיפור בעזרת סימן ירוק, ניתן לסמן כך רק עמוד אחד בסיפור.
* **Small, Wide, Tall, Large** – דרכי תצוגה של העמוד. המשתמש יכול לבחור את דרך התצוגה שהכי מתאימה לו בהתאם לארבעת האפשרויות הנ"ל.

****

**סימון העמוד ההתחלתי**

## תוכן העמוד

לאחר שלחצנו על אפשרות 2, אפשרות העריכה יפתח חלון חדש. החלון כולל אפשרות יציאה ואפשרות להגדלת/ הקטנת החלון, בנוסף החלון מחולק ל-3 חלקים עיקריים:



**(2)**

**(1)**

**(3)**

**חלק ראשון:** כותרת העמוד. נותנת לנו רמז לתוכן העמוד ומאפשרת לנו לגשת לעמוד מסוים מעמוד אחר. ניתן להחליף את הכותרת פשוט כאשר לוחצים עליה.

**חלק שני:** תגיות, ניתן להוסיף לכל עמוד תגיות שונות, על מנת להבדיל אותו מעמודים אחרים או על מנת ליצור מכנה משותף בין מספר עמודים שונים שיהיו בעלי תגית זהה.

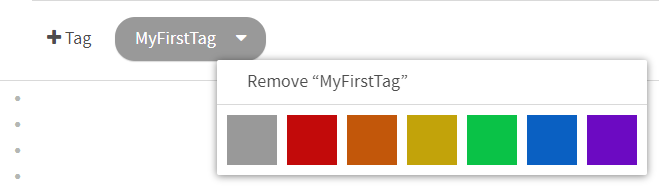
**חלק שלישי:** תוכן העמוד, נוכל לכתוב בו שיחה והשתלשלות אירועים, בנוסף נוכל להעביר את המשתמש לעמוד אחר בהתאם להחלטותיו ולבחירותיו. כאשר נמחק את שורת התוכן הכתובה כברירת מחדל, נקבל הוראות מהתוכנה על איך לכתוב טקסט בצורה נכונה ואיזה אפשרויות עומדות בפנינו כאשר אנו כותבים את תוכן העמוד.

### הוספת תגיות

על מנת להוסיף תגית עבור עמוד כלשהו נלחץ על **+ Tag** תפתח לנו האפשרות של להוסיף שם לתגית

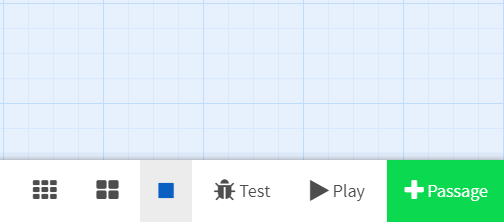
לאחר שבחרנו שם יש ללחוץ על **✓Add**.

לאחר שהוספנו תגית, ניתן למחוק אותה בעזרת לחיצה עליה תופיע לנו האפשרות של Remove “name” עם שם התגית שאותה אנו מעוניינים למחוק במקום name. בנוסף, יש אפשרות לשנות את צבע התגית , ישנם 7 צבעים לבחירה.



## תצוגה

כאשר יוצרים את הסיפור ישנן 3 דרכי תצוגה אפשריות שבעזרתן אנו יכולים לראות את העמודים בסיפור שלנו. האפשרויות נמצאות בתחתית המסך בצד ימין:



**(2)**

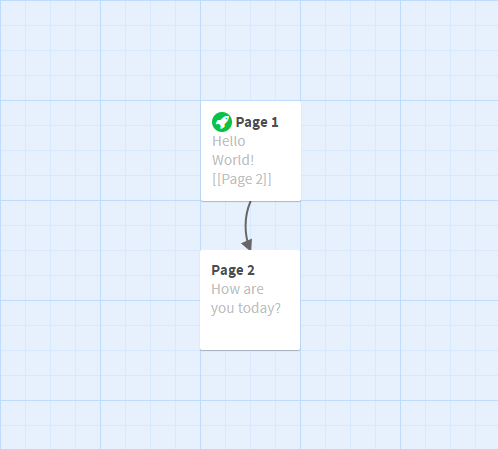
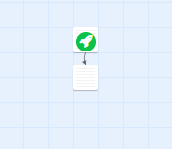
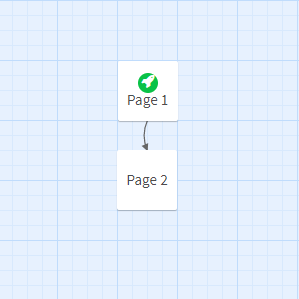
**(3)**

**(1)**

**אפשרות ראשונה:** האפשרות ההתחלתית, באפשרות זו ניתן לראות את הכותרת של העמוד ואת הטקסט רשום בו.

**אפשרות שנייה:** באפשרות זו ניתן לראות אך ורק את הכותרת של העמוד.

**אפשרות שלישית:** באפשרות זו לא ניתן לקבל מידע על תוכן העמוד או על הכותרת, אפשרות זו מראה לנו רק את מבנה הסיפור.

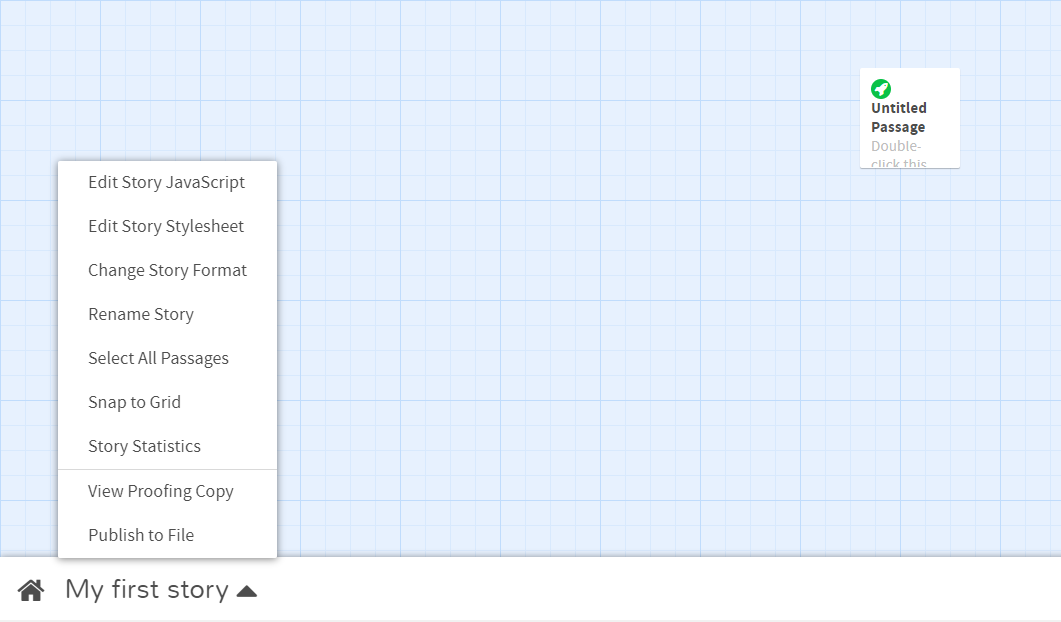


**(3)**

**(2)**

**(1)**

בעזרת העכבר ניתן למקם את עמודי הסיפור איפה שנרצה, לשימוש נוח יותר, כך שהעמודים ימוקמו אך ורק על המשבצות יש לעשות את השלבים הבאים:

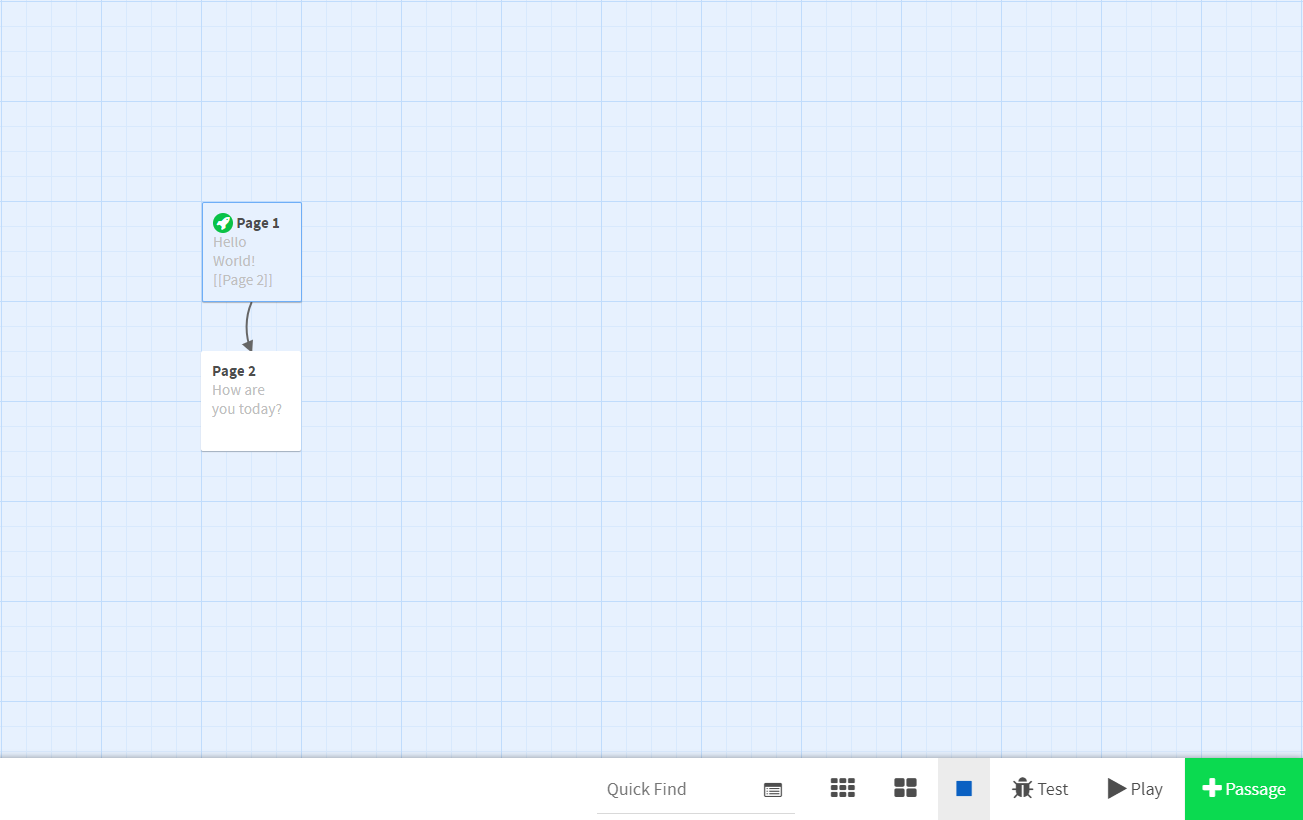


**בשלב הראשון:** נלחץ על החץ שליד השם שבחרנו לסיפור שלנו, נמצא בפינה שמאלית למטה. החץ יפתח לנו מספר אפשרויות שונות.

**בשלב השני:** נלחץ על Snap to Grid לאחר שנלחץ עליו יסומן לנו לידו וי. הסימון יאפשר לנו להזיז ולמקם את העמודים בצורה נוחה ומסודרת יותר.

# Debug and Play

כמו בכל תוכנה לכתיבת קוד, גם כאן יש לנו שתי אפשרויות לבדיקת התוצר שאנו כותבים. כאשר לוחצים על הפעלת ה Debug או על Play התהליך מתחיל מהעמוד הראשון של הסיפור.

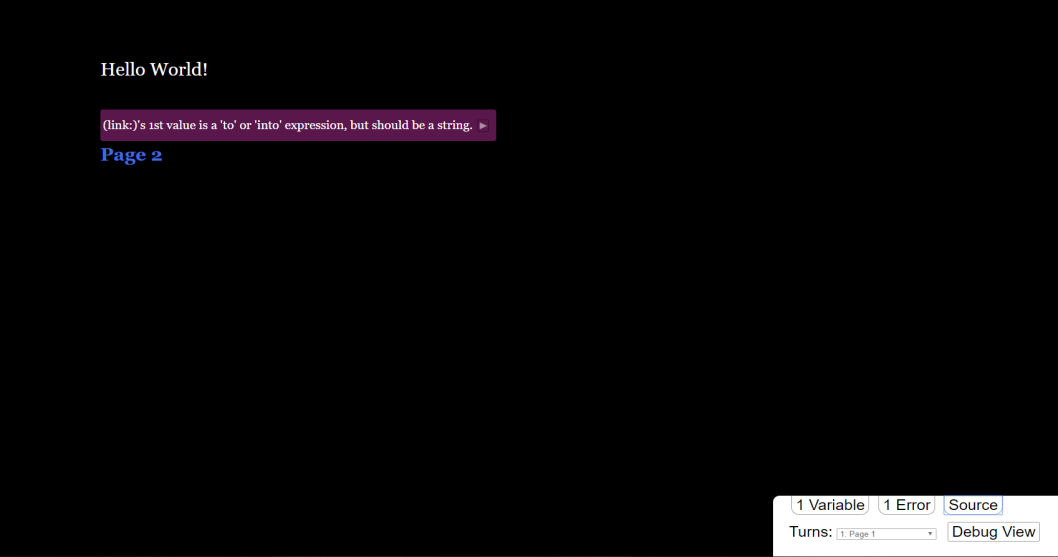


במידה והמשתמש עבר בין העמודים חץ חזרה המאפשר למתכנת או למשתמש לחזור לעמוד הקודם. על מנת למנוע מהמשתמש אפשרות זאת ניתן לבטל אותה, עוד מידע ניתן למצוא [כאן](#_העלמת_חץ_החזרה).

## Debug

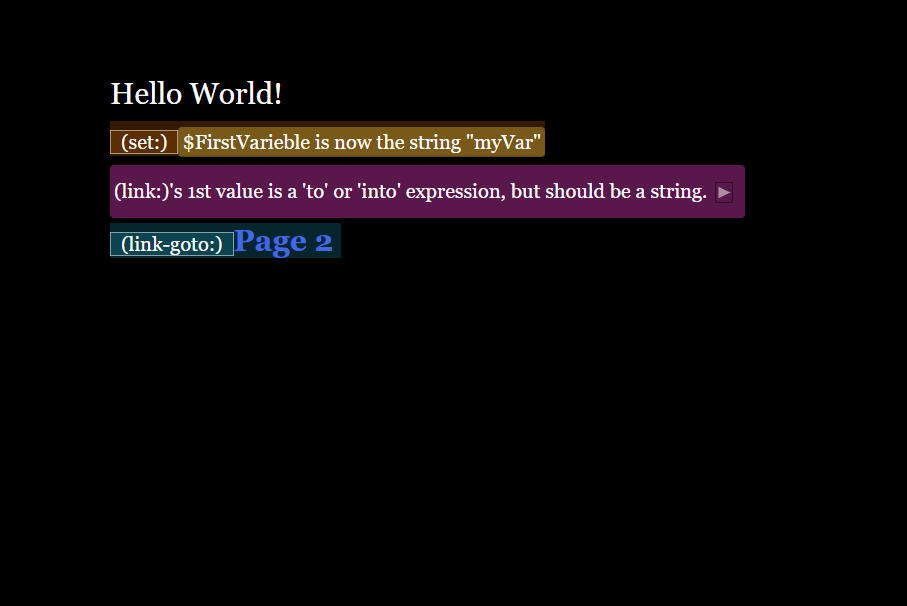
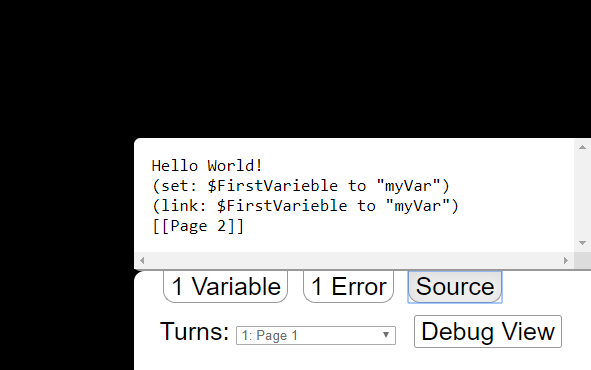
תהליך שיטתי של איתור והפחתת באגים (תקלים) בתוכנת מחשב או בחומרה אלקטרונית, כדי לגרום להם לפעול כפי שתוכננו. ככל שמתקיימים יותר קשרים בין תת-מערכות או שהמערכת חסרת סדר, כך תהליך ניפוי השגיאות יהיה קשה יותר. התהליך דומה מאוד ל Play למעט מספר שינויים קטנים הנמצאים בתחתית המסך מצד ימין. בדוגמא למטה נעבור על תהליך בדיקת השגיאות ונראה כיצד אנו יכולים לבדוק את המשתנים שהגדרנו, טעויות, קוד המקור והדף הנוכחי בו אנו נמצאים.

הכנסנו קוד פשוט לעמוד הראשון שיצרנו על מנת להדגים את כל האפשרויות שתהליך מציע למשתמש.



**Source** – במידה ונלחץ עליו, מאפשר לנו לראות את השורות שאותן כתבנו בתוכן העמוד.

**Debug View** – במידה ונלחץ עליו, מאפשר לנו לראות את השורות שאותן כתבנו בתוכן העמוד ולאיזה שורה הן מתייחסות בשורות שהמשתמש רואה. דרך נוספת לראות שורות אלו בצורה נוחה יותר היא לא להדליק את ה Debug View אלה פשוט להעביר את העכבר על השורה מבלי ללחוץ עליה ואז יופיע הקוד שהמתייחס לשורה זו.



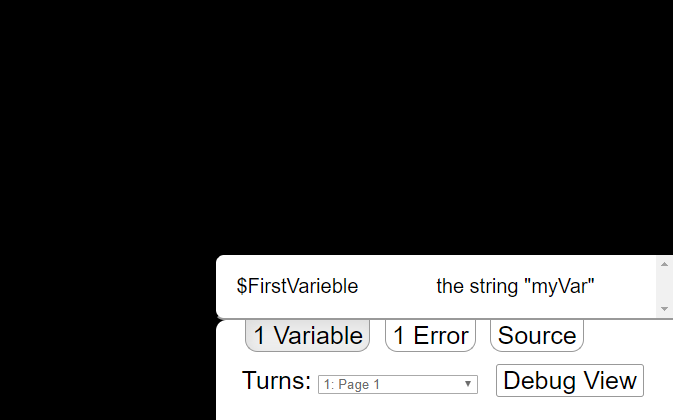
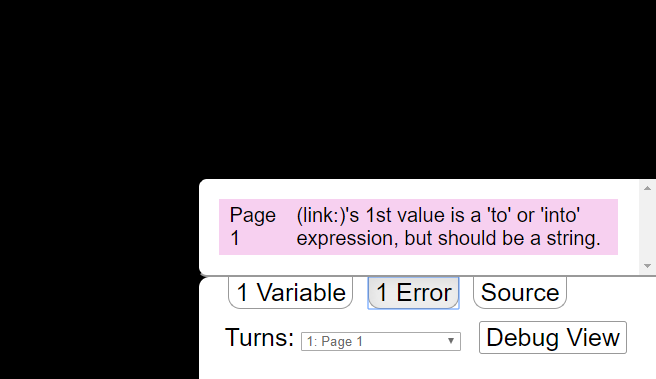
**Source**

**Debug View**

**Error** – לפני המילה מופיע לנו מספר השגיאות בסיפור, במקרה זה יצרנו שגיאה מכוונת על מנת להראות איך הדיבאגר מסמן למתכנת על באגים בקוד. במידה וקיימת שגיאה בקוד הדיבאגר יסמן אותה ברקע סגול ויכתוב משוב על מקור השגיאה. במקרה שלנו הגדרנו קישור בצורה שגויה.

**Variable** – לפני המילה מופיע לנו מספר המשתנים שהגדרנו בסיפור, הוא מראה מה שם המשתנה שהוגדר ומה הערך שנתנו לו. במקרה זה בסיפור שלנו הגדרנו רק משתנה אחד בשורה השנייה של הקוד בעזרת הפקודה :set עוד על הגדרת משתנים יהיה ניתן לראות [כאן](#_משתנים).

**Turns** – מאפשר לכנו לראות את העמוד שבו אנו נמצאים.



**Error**

**Variable**

## Play

מדמה את המשחק מנקודת המבט של השחקן. נראה בדיוק כמו הדיבאגר למעט חוסר האפשרויות ובדיקות הנמצאות בצד ימין. כלומר, כן יראה שגיאות במידה ויש אך לא יראה לנו את הכמות הכללית של השגיאות או את שאר המידע שאנו מקבלים דרך הדיבאגר.

****

# שפות תכנות

Twine תומך בשתי שפות תכנות מוכרות: HTML ו- JavaScript.

## HTML

ניתן לכתוב קוד בשפת HTML ישירות ב[תוכן העמוד](#_תוכן_העמוד) שאנו יוצרים. ניתן לכתוב קוד בצורה הבאה בתוך תוכן העמוד:

<mark>This is marked text.

&para; So is this.

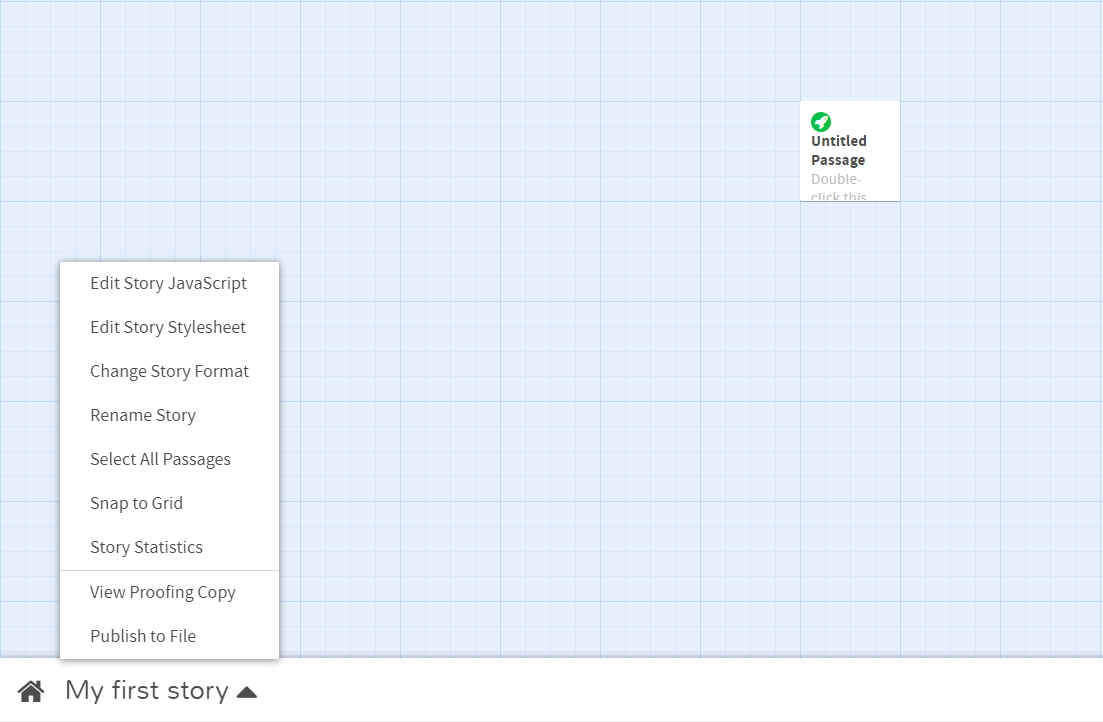
And this.</mark>

נשתמש בעיקר כשנרצה להגדיר תמונות בעמוד, אך ניתן להשתמש בנושאים רבים ומגוונים. עוד מידע על הוספת תמונות [כאן](#_הוספת_תמונות).

בנוסף, ניתן להוסיף גם תגית <script> ובתוכה להריץ קוד של JavaScript בתוך תוכן העמוד.

HTML מאפשר למתכנת להשתמש ב תגית <!-- Comment --> המאפשרת הוספת הערה על הקוד למקרה שהקוד לא ברור בשלב מתקדם יותר של התכנות.

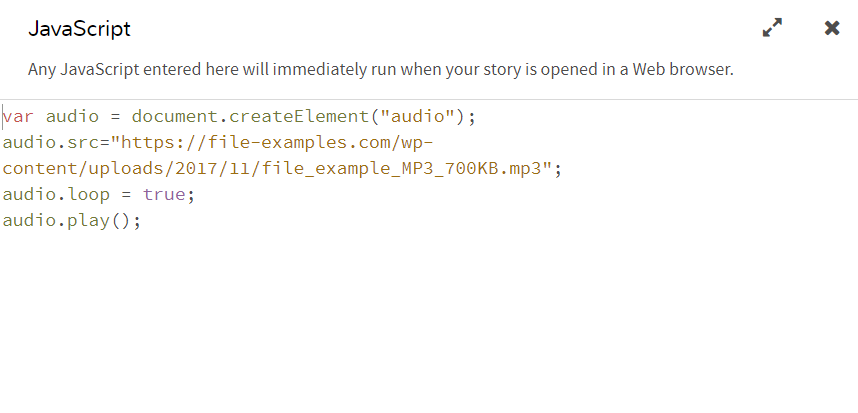
## JavaScript

מאפשר שימוש בסקריפטים שונים, המוסיפים עומק ורבדים חדשים למשחק, דוגמא לרובד כזה הוא הוספת מוזיקה לעמוד המשחק בעזרת סקריפט.

בתמונה למעלה אנו רואים אפשרות הוספת סקריפט זהה לכל העמודים במשחק. ניתן לכתוב סקריפט אינדיבידואלי לכל עמוד בעזרת תגית <script> שהוזכרה בחלק של ה HTML.

### הוספת מוזיקה לעמוד המשחק

על מנת לעשות את המשחק יותר מעניין ויותר מושך עבור המשתמש, נרצה להוסיף לו רובד נוסף – מוזיקה. ניתן להוסיף מוזיקה בעזרת script. בשלב הראשון נפתח את Edit Story JavaScript כפי שאפשר לראות בתמונה למעלה. לאחר מכן יפתח לנו חלון חדש בו נוכל לכתוב את הסקריפט שלנו. לפני שנכתוב את הקוד עלינו למצוא קובץ שמע באינטרנט ולהעתיק את הקישור לקובץ עצמו (להעתיק את ה URL), חשוב לוודא שאכן העתקנו את הקישור לקובץ עצמו, ניתן לבדוק זאת אם הקישור נגמר ב .mp3



בדרך זו אנו מגדירים את אותה המוזיקה לכל חלקי הסיפור שלנו.

כעת, נעבור על הסקריפט ונסביר את הקוד:

**בשורה הראשונה** – אנו יוצרים את משתנה השמע.

**בשורה השנייה והשלישית** – אנו מגדירים את המקור של המשתנה.

**בשורה הרביעית** – כאשר אנו מגדירים **audio.loop** = **true** זה אומר שאנו נחזור על אותו שמע בצורה אינסופית, במידה ולא היינו מכניסים את השורה הזו המוזיקה הייתה מתנגנת פעם אחת בלבד ומפסיקה.

**בשורה החמישית** – אנו מגדירים שהמוזיקה תתחיל להתנגן.

על מנת להוסיף מוזיקה בעמוד ספציפי יש לכתוב את הקוד הבא בתוך תוכן העמוד:

על מנת לבדוק שאכן הוספת המוזיקה עבדה, יש פשוט להריץ את הסיפור בעזרת Debug או Play.

# ערכית תוכן העמוד

## עריכת טקסט בסיסית

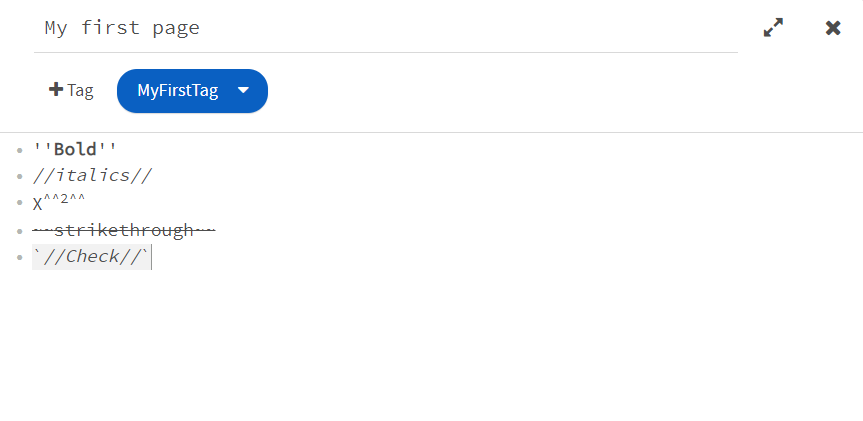
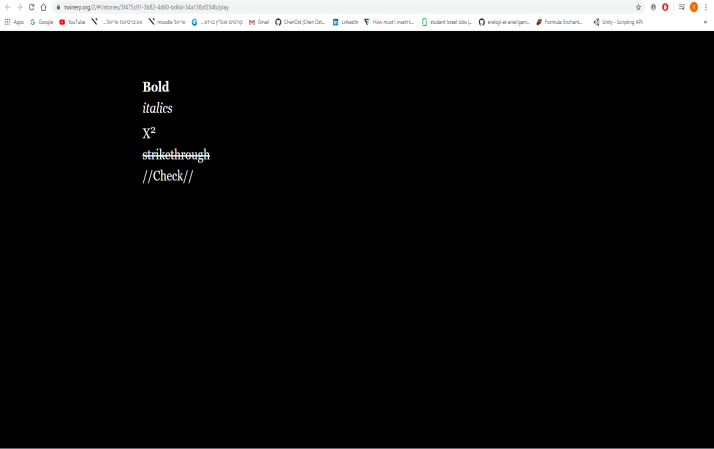
ניתן לכתוב בתוכן העמוד כל דבר שיתחשק לנו. ישנן אפשרויות רבות המאפשרות עריכה שונה של תוכן העמוד:

1. ''Bold'' – מסמן את המילה בין ''X'' ב**בולד X**.
2. //italics// – מסמן את המילה בין //X// ב*אלכסון X*.
3. ^^super^^ - מסמן את המילה בין ^^ X ^^ כחזקה x.
4. ~~strikethrough~~ – מסמן את המילה בין ~~X~~ כמילה עם קו באמצע ~~X~~.
5. `backticks` – על מנת להציג סמלים מיוחדים מבלי שהם ישתנו יש לשים אותם בין ` `.

ניתן להוסיף לטקסט עוד אפקטים רבים: אפשר למצוא עוד מידע [כאן](https://twine2.neocities.org/#macro_text-style).

**דוגמאות קוד:**

בכל שורה כתבנו אפשרות שונה בהתאם לסדר כפי שצוין למעלה.



## הוספת קישורים

ב Twine קיימים מספר סוגים שונים של קישורים (Links) אחד הסוגים העיקריים הוא קישור שניתן לכתוב בתוכן העמוד אשר ישלח אותנו לעמוד אחר, כלומר ניתן ליצור Link מעמוד אחד לעמוד אחר בהתאם לבחירת המשתמש. ניתן ליצור קישור לעמוד שכבר קיים אך במידה ואנו יוצרים קישור לעמוד שטרם קיים, התוכנה תיצור את העמוד עם השם כפי שהעברנו אותו. ישנן 3 דרכים לבצע זאת:

1. [[link text -> passage name]] – ה link textהוא הטקסט שאותו המשתמש רואה ובוא הוא בוחר, לאחר שהמשתמש בחר אפשרות זאת הלינק מועבר ל passage name בעזרת הכותרת של העמוד, כלומר ב passage name נרשום את הכותרת של העמוד אליו אנו רוצים לעבור.
2. [[link text <- passage name]] – אותו דבר כמו באפשרות מספר 1.
3. [[passage name]] המשתמש רואה את הכותרת של העמוד אליו הוא מועבר ללא טקסט אחר שנכתב במיוחד.

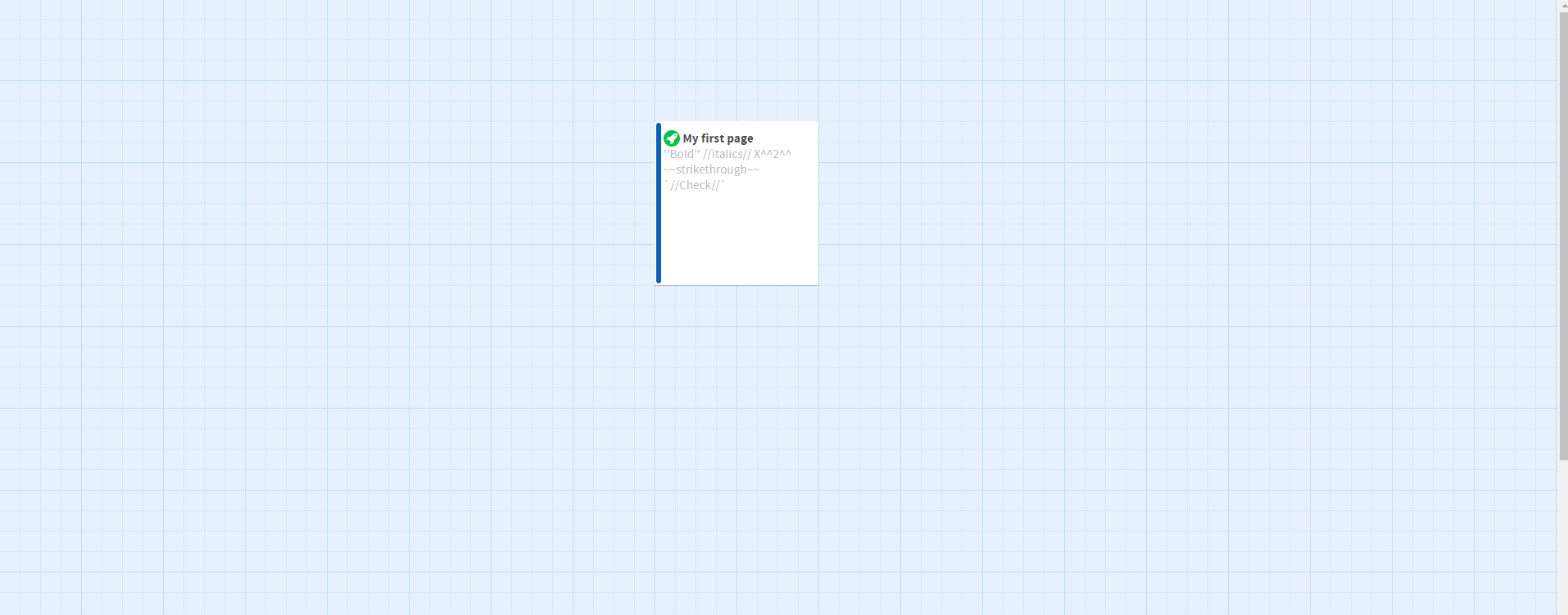
להרחבה בנושא קישורים לעמודי אחרים ניתן למצוא [כאן](https://twine2.neocities.org/#markup_link).

מעבר לסוג הנ"ל, לקישורים יש עוד תכונות רבות, שניתן לקרוא עליהן בהרחבה [כאן](https://twine2.neocities.org/#macro_text-style).

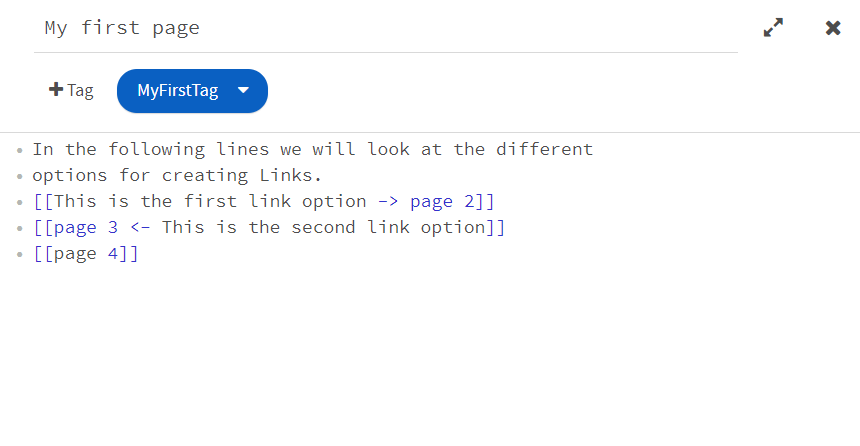
**דוגמאות קוד:**

נעבור ליצירת קישורים לעמודים חדשים. על מנת לראות שאכן נוצרים לנו עמודים חדשים תוך כדי כתיבת שורות הטקסט,

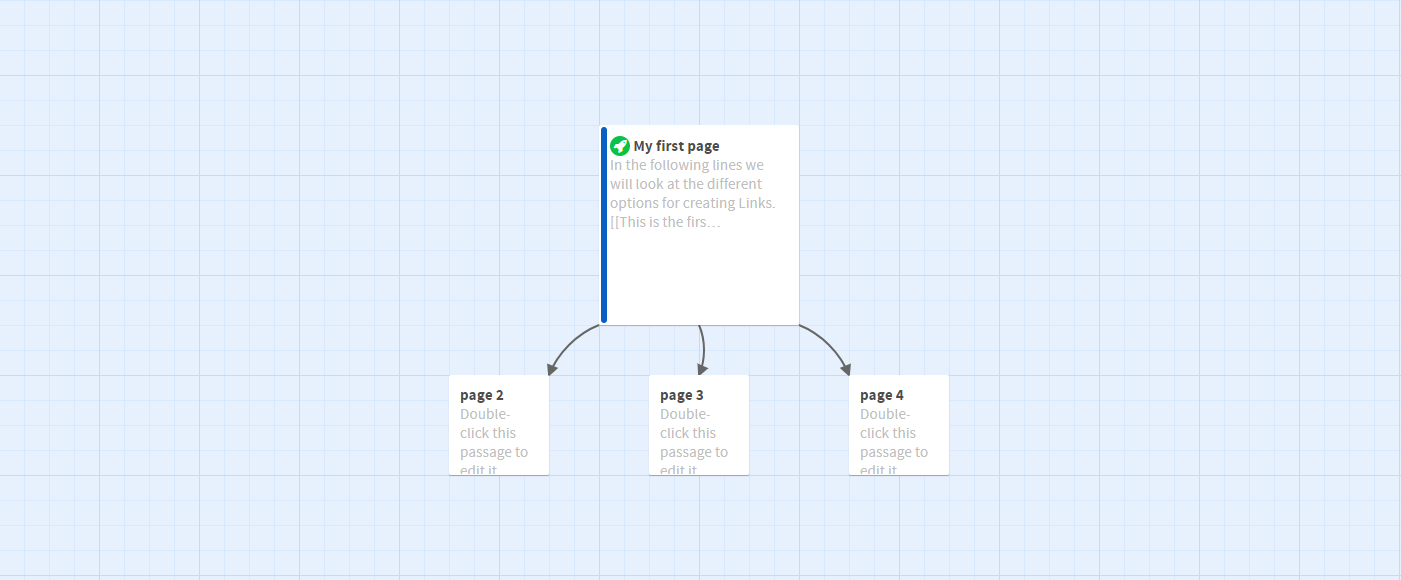
העמוד הראשון מכיל את המידע כפי שניתן לראות בתמונת צילום מסך מהעמוד הקודם, בתמונה הבאה שינינו את תצוגת העמוד ל Large על מנת שנוכל לראות את תוכן העמוד מהמסך הראשי בבירור. קל לראות שלעמוד אין קישור לעמודים נוספים ולמעשה לא קיימים עמודים חוץ מעמוד אחד.



כעת, נמחק את תוכן העמוד ונכתוב במקומו את כל שלושת האפשרויות של הLinks.

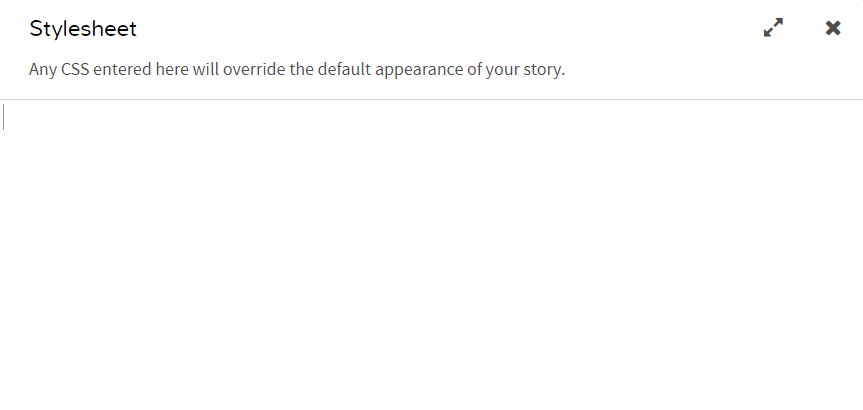
****

מכאן נעבור למסך הראשי של הסיפור ונראה כיצד הוא השתנה בהתאם לקישורים החדשים שיצרנו.



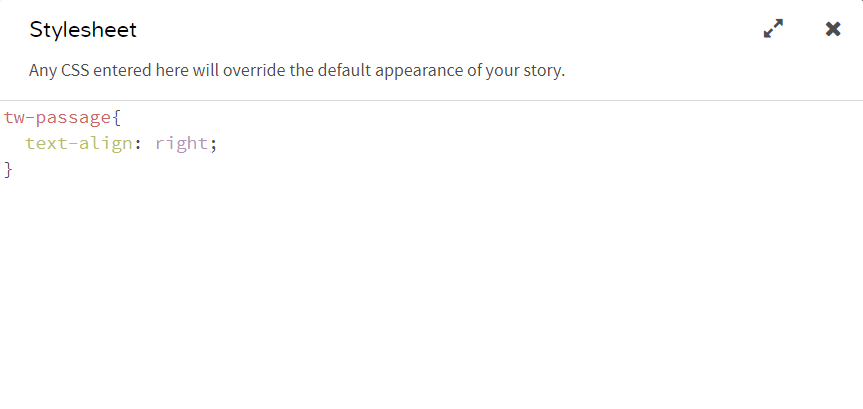
ניתן לראות שנוצרו לעמוד המקורי שלנו עוד שלושה עמודים המקושרים אליו, כאשר כל עמוד שנוצר נוצר עם ערכים התחלתיים, לכן כל שלושת העמודים מכילים את אותם נתונים למעט הכותרת שאותה העברנו בעצמנו. על מנת לבדוק שאכן הקישורים עובדים נעבור לאחד המצבים Debug או Play. במצב שבחרנו נלחץ על הקישור ואכן נראה שאנו עוברים לעמוד הבא.

## Edit Story Stylesheet

לאחר שנלחץ על Edit Story Stylesheet יפתח לפנינו חלון חדש, בחלון זה אנו יכולים לשנות את כל הערכים ההתחלתיים שניתנים לסיפור שלנו מבחינת נראות. הערכים שאנו כותבים בחלון זה משפיעים על כל העמודים בסיפור שלנו באותה דרך, על מנת לשנות עמוד בצורה אינדיבידואלית עלינו לכתוב קוד ספציפי בתוכן העמוד.

## יישור טקסט וכתיבת טקסט בעברית

צורת ברירת המחדל שבה נכתב הטקסט הוא משמאל לימין, אך התוכנה מאפשרת שימוש בשפות שונות, כולל עברית, על מנת שהסדר ישמר גם כשאנו כותבים בשפה מימין לשמאל ולא משמאל לימין, עלינו ליישר את הטקסט בהתאם לשפה שבה אנו כותבים. בנוסף, ניתן ליישר גם את הטקסט לאמצע המסך (לדוגמא, כמו שעושים עם כותרות) על מנת ליישר את הטקסט עלינו לגשת ל:

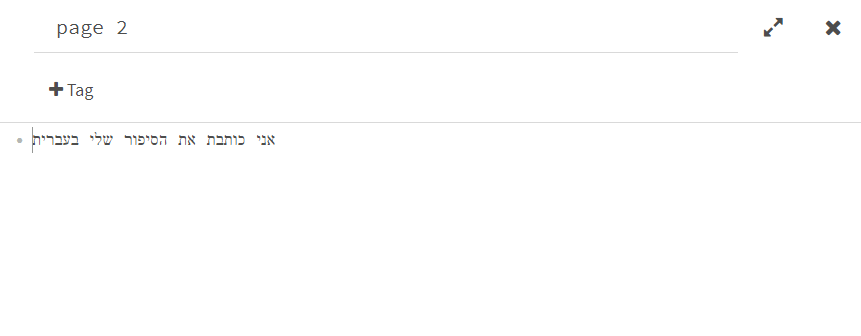
Edit Story Stylesheet. על מנת לשנות את יישור הטקסט נכתוב את הקוד הבא:

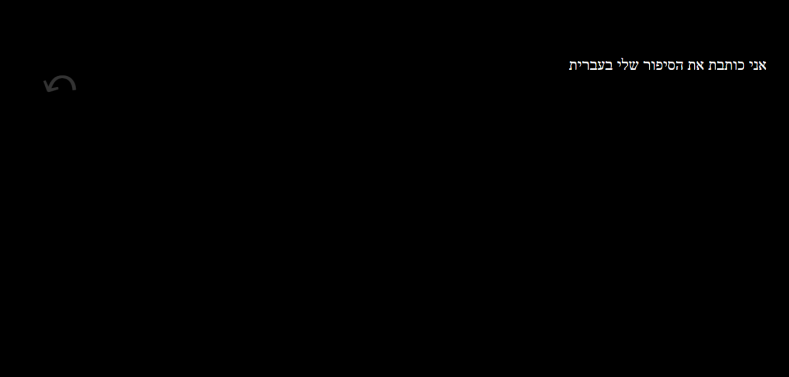
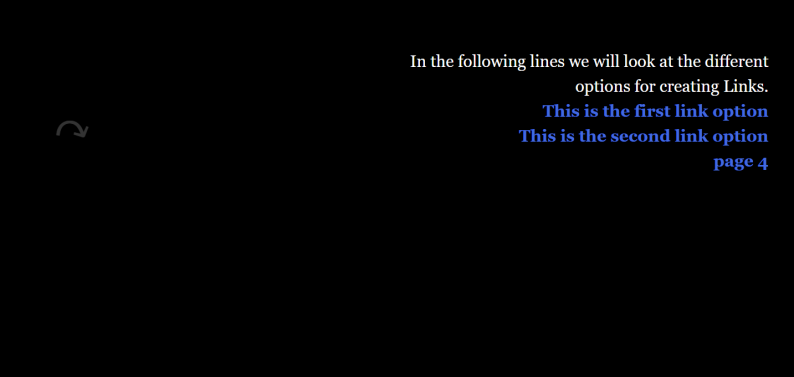
**tw-passage** אומר שהקטע קוד בסוגריים המסולסלים ישפיע על כל התוכן של העמודים.

**text-align** הפעולה שאנו מבצעים על הטקסט, במקרה שלנו יישור השורה לכיוון שאותו נבחר.

**right** הכיוון שאליו בחרנו ליישר את הטקסט, באותה מידה יכולנו לבחור גם: center, left. למרות ש left הוא הערך הדיפולטיבי של התוכנה.

כעת, נעשה בדיקה שאכן הקוד שכתבנו עובד, בדף מספר 2 נרשום טקסט בעברית:



בתמונה מצד ימין מופיע העמוד הראשון עם קישור לעמוד מספר 2, לאחר שנלחץ על הקישור נגיע לעמוד 2 הנמצא בצד שמאל.

## שינויים לעמודים ספציפיים

בדוגמא הבא נחזור על הדוגמא מהנושא הקודם, ניתן לראות שהקוד אכן עובד, אבל האם הוא עובד בדיוק כמו שרצינו?

התשובה היא לא, משום שהוא גם מיישר לימין טקסט שנכתב באנגלית. כיצד ניתן לטפל במקרים כאלו ולעשות הפרדה?

על מנת לעשות את ההפרדה נוסיף לעמוד 2 את התגית “Hebrew” ונשנה את הקוד שלנו בצורה הבאה:

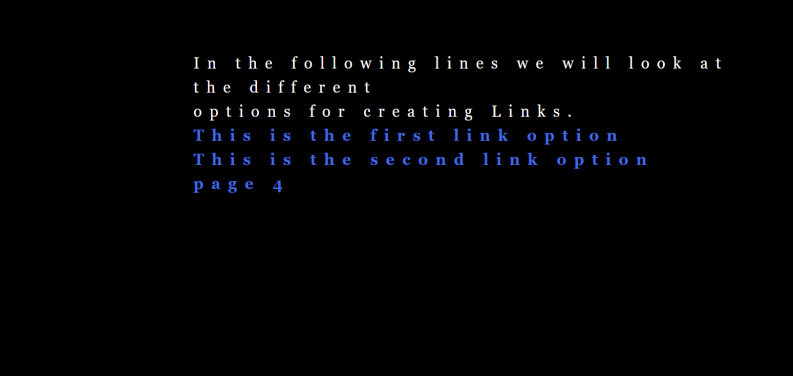
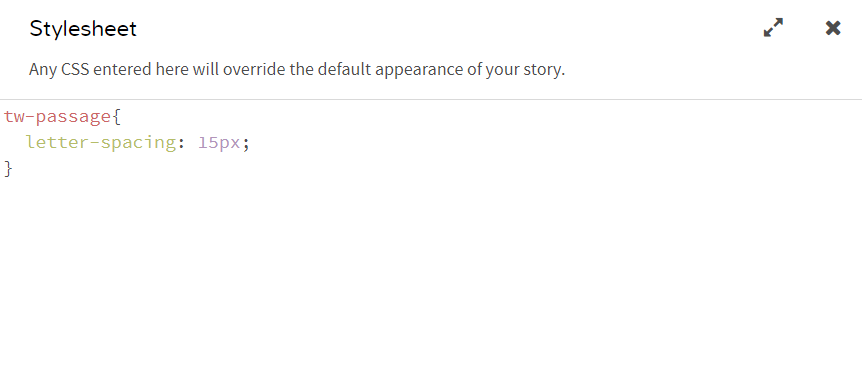


**[ tags = “Hebrew”**] הוספנו לקוד אפשרות שרק תגיות שהשם שלהן הוא Hebrew יעשו את השינוי שכתבנו, ואכן לאחר בדיקה נוספת התנאי הזה עובד.

בהמשך נראה דוגמאות רבות של שינויים אשר בכולם שיטת העבודה שהראנו תתאים, כלומר על מנת לבצע שינוי על לקבוצה ספציפית של עמודים, כל מה שצריך לעשות הוא להוסיף לעמודים שבהם נרצה לעשות שינוי תגית מותאמת ואחידה. לאחר מכן נוסיף לקוד לפני הסוגריים המסולסלים  **[ tags = “name”**]כאשר name הוא שם התגית שבחרנו.

## רווח בין אותיות

כאשר נשתמש בפונטים שונים של כתב, נתקל שבחלק מהפונטים האותיות מאוד צמודות. על מנת לתקן בעיה ולהוסיף רווח בין האותיות, נכתוב קוד ב Edit Story Stylesheet. אין צורך למחוק את הקודים שכתבנו בתרגילים הקודמים ניתן לכתוב מספר דברים תחת אותן סוגריים מסולסלות, כאן מחקנו על מנת להראות בצורה נוחה את השינויים החדשים שנוספו.

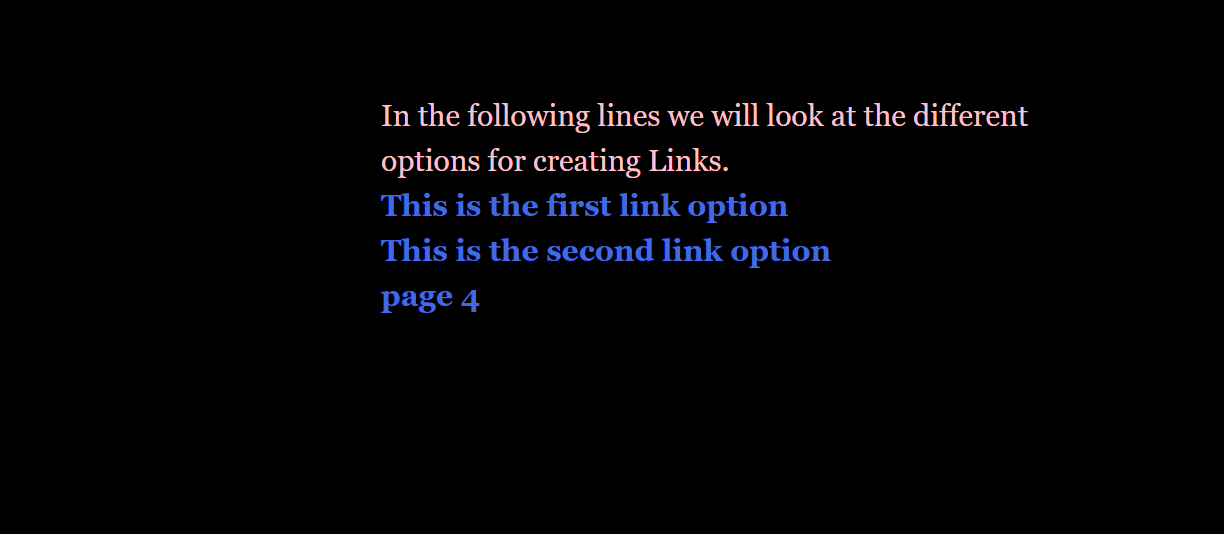
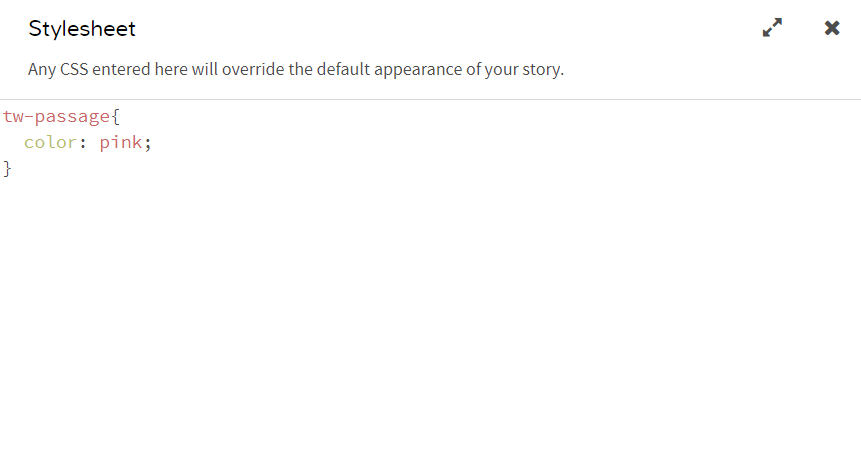


**letter-spacing** הפעולה שאנו מבצעים על הטקסט, במקרה שלנו הוספת רווח בין האותיות.

**15px** כמות הפיקסלים המפרידים בין אות לאות. ניתן לשנות את המספר 15 לכל מספר אחר, בדוגמא זו השתמשנו במספר זה על מנת להדגיש את הרווחים.

## שינוי צבע הטקסט

במידה ונרצה לשנות את צבע הטקסט בסיפור שלנו, נוכל להשתמש בקוד הבא:



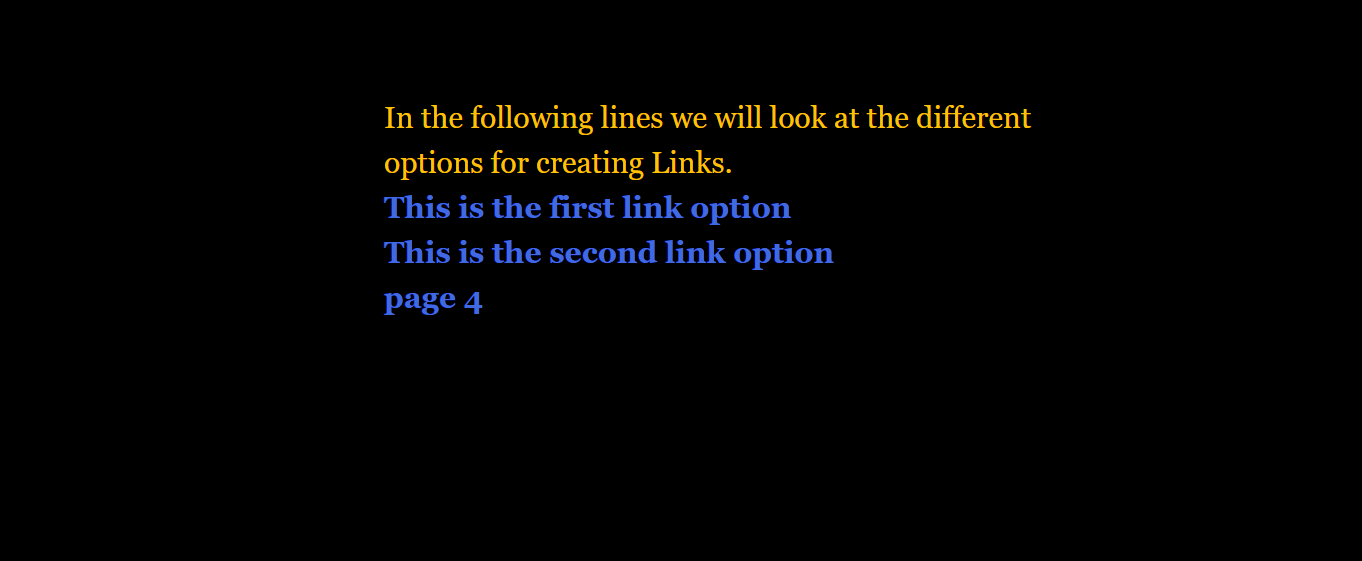
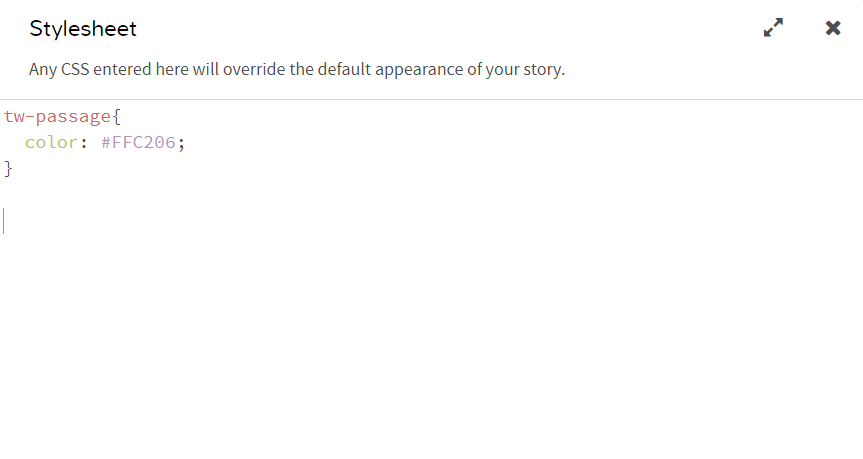
בדוגמא למעלה, השתמשנו ב **color** על מנת לשנות את צבע הטקסט, נשים לב שזה שינה רק את צבע הטקסט ולא את צבע הקישורים, דבר נוסף שצבעו שונה זה את הצבע של החץ שחוזר אחורנית. בדוגמא השתמשנו בשיטה אחת מתוך שתיים לשינוי צבע,

לאחר ה **color** ניתן לכתוב 2 דברים:

**השיטה הראשונה –** לכתוב שם של צבע: <https://www.w3schools.com/colors/colors_names.asp> רשימה של כל שמות הצבעים שנתמכים ב HTML.

**השיטה השנייה –** לכתוב ערך של צבע: <https://htmlcolorcodes.com/> ניתן למצוא גוון של צבע ולהעתיק את המספר שלו, חשוב לשים לב להעתיק # לפני.

שתי השיטות רלוונטיות לכל שינוי צבע נבצע ולא רק עבור שינוי צבע טקסט.

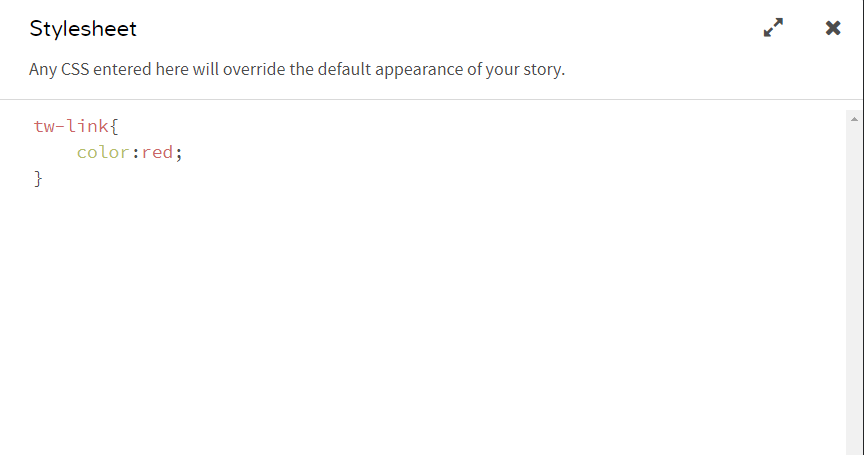
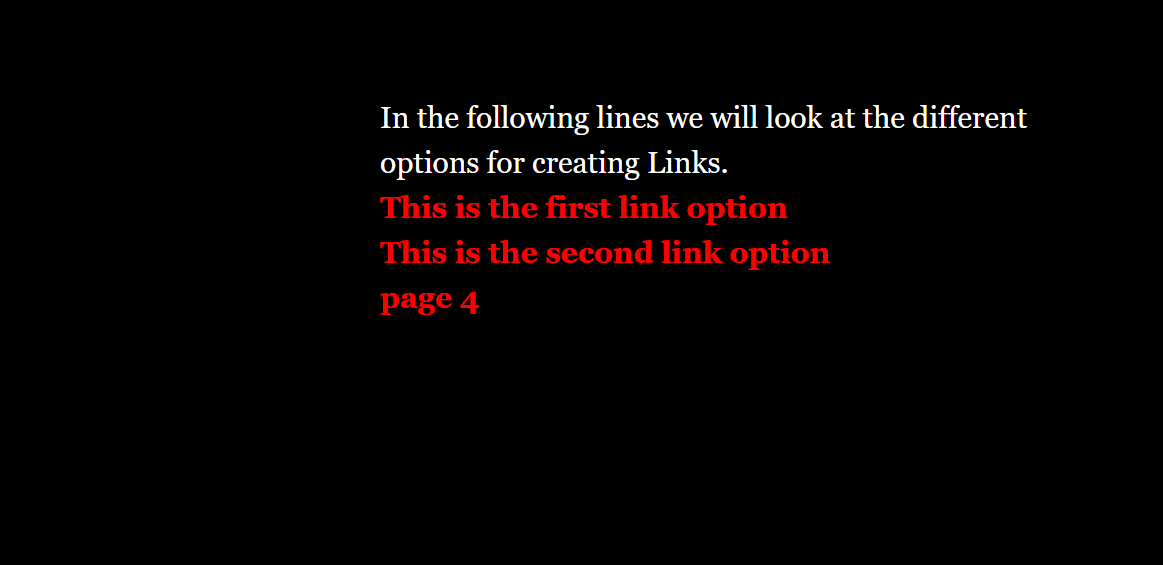


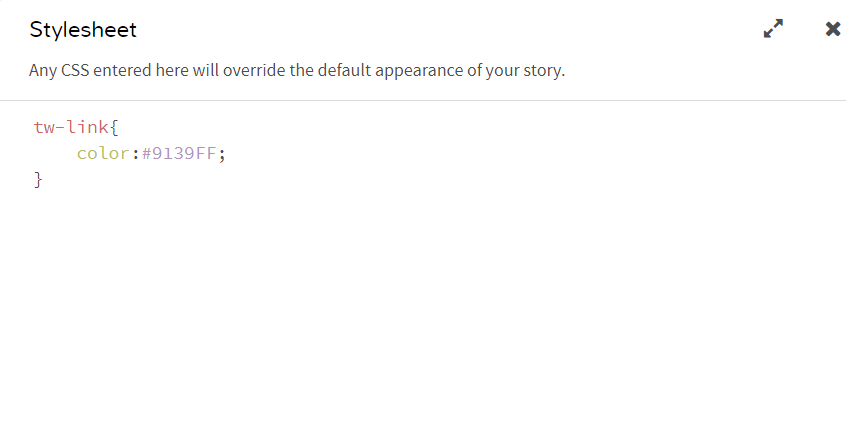
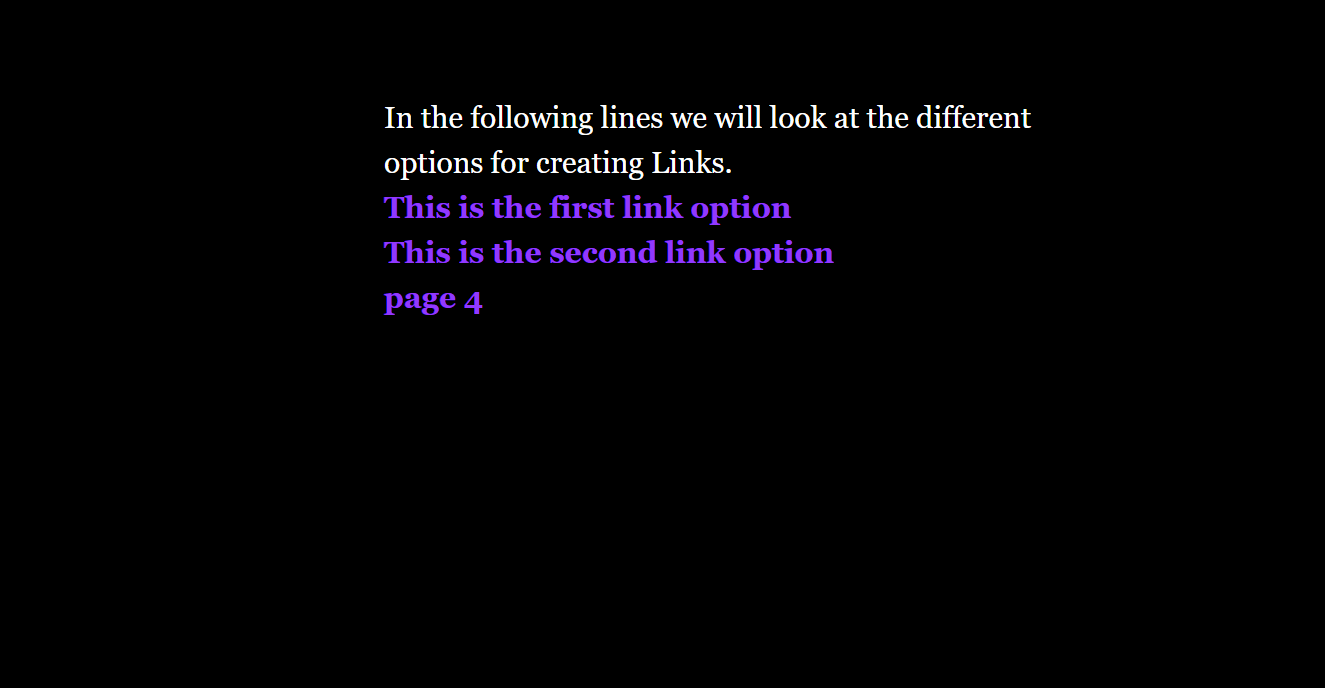
## שינוי צבע הלינק

על מנת לשנות את צבע הלינק (הקישורים) בעזרתם אנו עוברים בין העמודים השונים, עלינו לשנות את השורה בקוד המתייחסת לעמודים ולשנות אותה שתתייחס ללינקים (קישורים). כלומר נחליף את **tw-passage** (אומר שקטע הקוד בסוגריים המסולסל

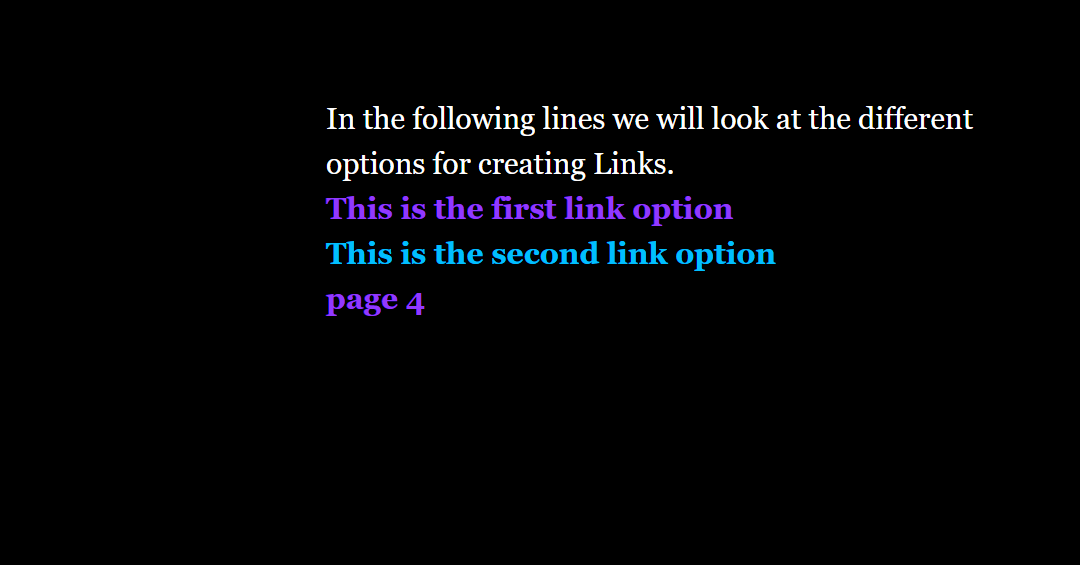
ים ישפיע על כל העמודים) ל **tw-link** אומר שהקטע קוד בסוגריים המסולסלים ישפיע על כל הלינקים.

גם כאן ישנן 2 דרכים לשנות את צבע הלינק ( ראו ב- שינוי צבע טקסט).

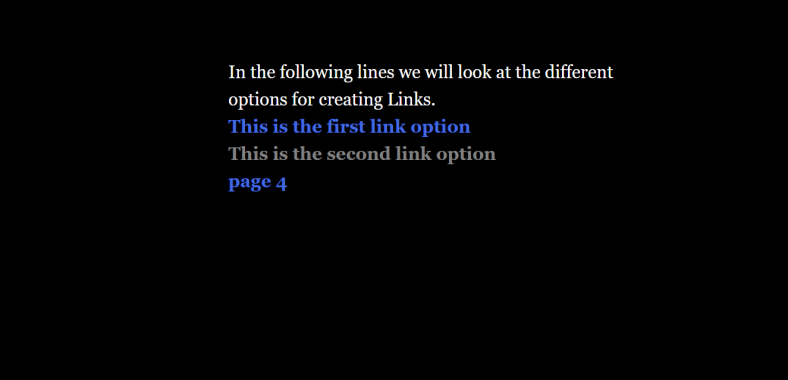
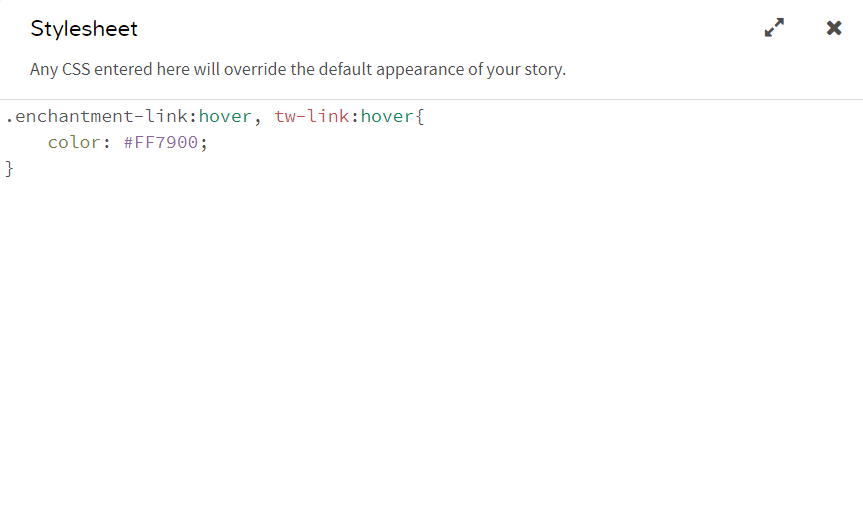
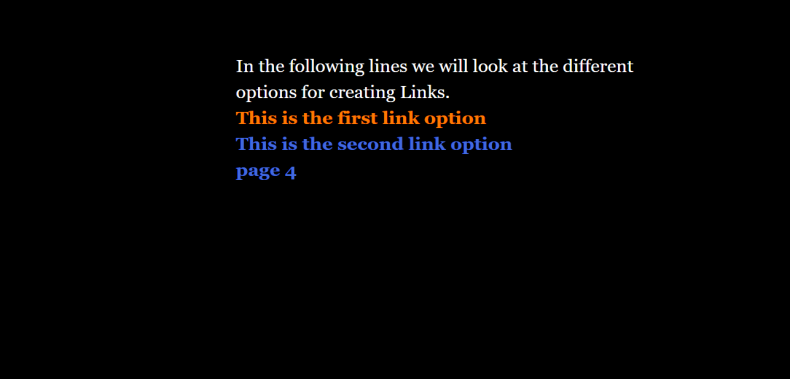




כאשר מעבירים את העכבר על הקישור הוא יצבע בצבע תכלת, צבע זה מופיע גם כאשר צבע הקישור הוא צבע ברירת מחדל אך פחות שמים לב להבדל כי צבע ברירת מחדל של הקישור הוא כחול.



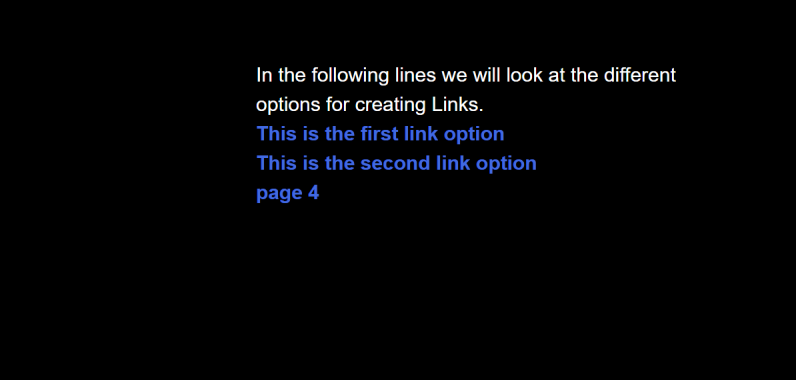
על מנת לשנות את צבע ההקלקה של הקישור נשתמש בקוד הבא:



גם כאן ניתן להשתמש בשתי השיטות שעברנו עליהן מקודם

הסבר על הקוד סירטון 6

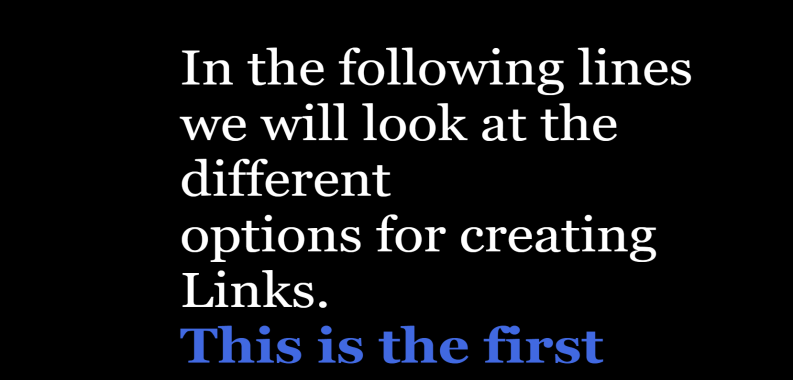
## גופן הטקסט

כאשר נרצה לשנות בכל העמודים את גופן הטקסט נשתמש בקוד הבא:

חשוב לשים לב, לפני שנבחר את סוג הגופן שבו אנו מעוניינים להשתמש חשוב לוודא שאכן המחשב שלנו מכיר את סוג הגופן הזה, כלומר הגופן חייב להיות מותקן אצלנו במחשב.

## שינוי גודל הטקסט

על מנת לשנות את גודל הטקסט בכל העמודים נשתמש בקוד הבא:



**font-size** הפקודה שבעזרתה אנו משנים את גודל הטקסט, בסוף הפקודה הזו יש לכתוב את המספר הקבוע באיזה גודל יהיה הטקסט (בדוגמא בתמונה בחרנו את המספר 6) וקיימות מספר אפשרויות שניתן לכתוב על מנת לקבוע גודל שונה בהתאם להחלטת המתכנת:

**vh** – משפיע על ה Height של הטקסט, כלומר על גובה הטקסט.

**vw** – משפיע על ה width של הטקסט, כלומר על רוחב הטקסט.

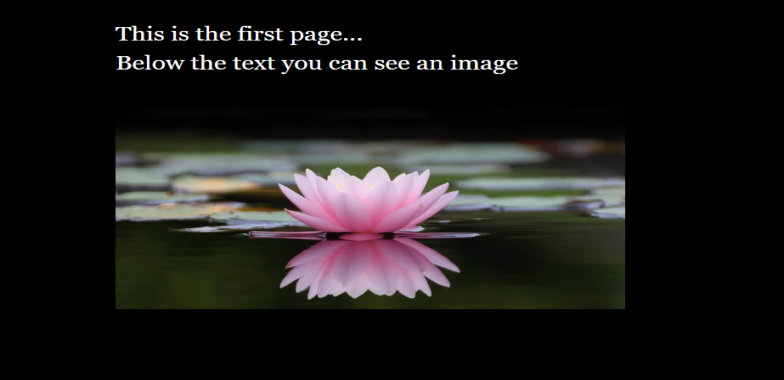
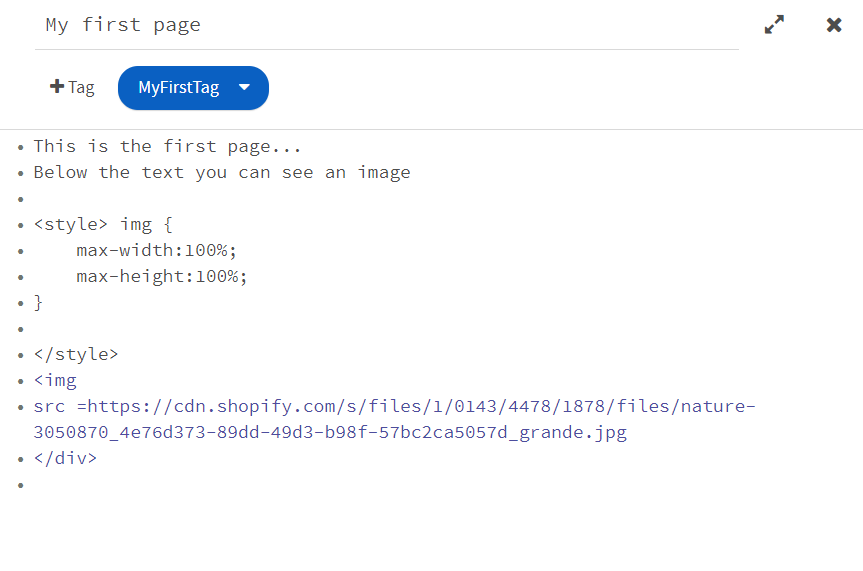
**vmin** – הגודל המינימלי של הטקסט.

**vmax** – הגודל המקסימלי של הטקסט.

חשוב לשים לב! לאחר שקבענו גודל שונה לטקסט שלנו יש לכתוב את השורה **line-height: normal**  שורה זו שומרת שבזמן שינוי גודל המסך הטקסט עדיין יהיה ניתן לקריאה והרווח בין השורות יישאר תקין.

## הוספת תמונות

במידה ונרצה להוסיף תמונה לאחד מהעמודים בסיפור עלינו להוסיף את התמונה בצורה אינדיווידואלית , כלומר להוסיף אותה ישירות לעמוד הרצוי ולא בעזרת ה Edit Story Stylesheets. כפי שכבר ראינו התוכנה Twine תומכת בשפת התכנות HTML, אחד המקומות שבהם שימוש בשפה זו בה לידי ביטוי הוא בעת הוספת תמונה לעמוד. לפני שנכתוב את הקוד עלינו למצוא תמונה מהאינטרנט ולהעתיק את הקישור לתמונה עצמה (להעתיק את ה URL), חשוב לוודא שאכן העתקנו את הקישור לתמונה עצמה, ניתן לבדוק זאת אם הקישור נגמר ב .png או .jpg



כעת, נסביר את הקוד:

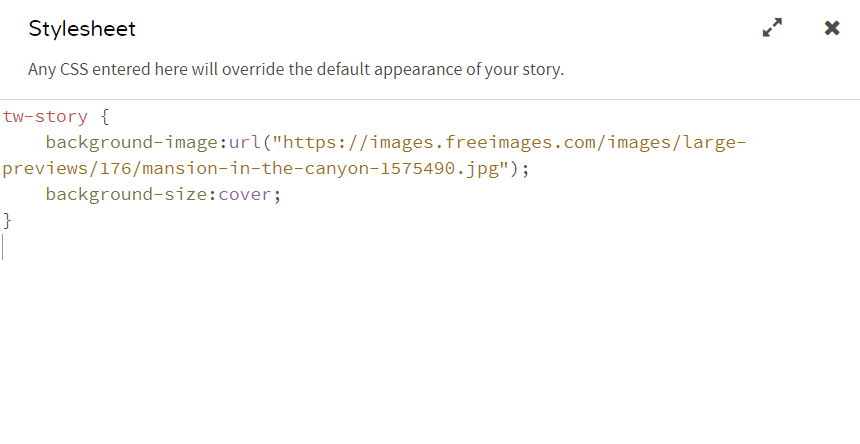
**שורות 4-9:** בין התגיות style אנו מגדירים את עיצוב התמונה, כלומר אנו מגדירים איך התמונה תראה מבחינת אורך ורוחב. על מנת שהתמונה לא תשתנה כאשר אנו מקטינים/ מגדילים את המסך או כאשר אנו משתמשים במכשירים שונים: מחשב, טלפון, טבלאט ועוד. כמו כן, ניתן להוסיף בין שורות קוד אלו עוד מרכיבים האחראים על עיצוב התמונה, ניתן למצוא עוד אפשרויות עיצוב [כאן](https://www.w3schools.com/css/css3_images.asp).

**שורות 10-12:** אנו מגדירים את התמונה עצמה, אנו מוסיפים את קישור ה URL שבחרנו, כפי שניתן לראות הקישור אכן נגמר ב .jpg כלומר, מדובר בקישור תקני של תמונה.

### הגדרת תמונת רקע

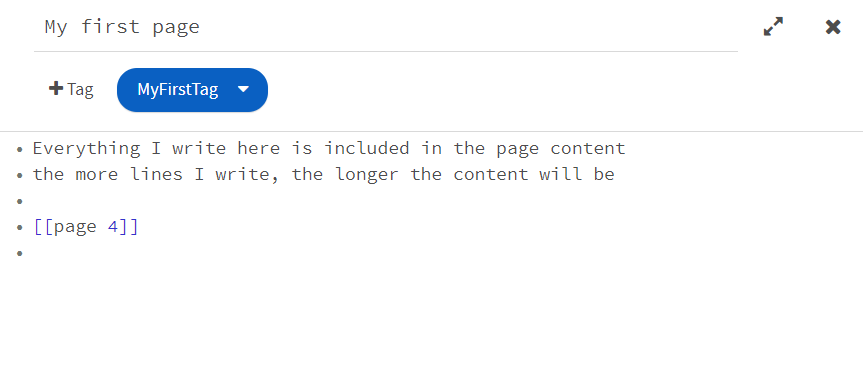
כאשר נרצה להגדיר תמונת רקע שתהיה זהה לכל העמודים הנמצאים אצלנו בסיפור עלינו לחזור ל Edit Story Stylesheets.

גם כאן ניתן להשתמש בתגיות שונות על מנת להגדיר תמונות רקע שונות עבור חלקים שונים בסיפור. לפני שנכתוב את הקוד עלינו למצוא תמונה מהאינטרנט ולהעתיק את הקישור לתמונה עצמה (להעתיק את ה URL), חשוב לוודא שאכן העתקנו את הקישור לתמונה עצמה, ניתן לבדוק זאת אם הקישור נגמר ב .png או .jpg

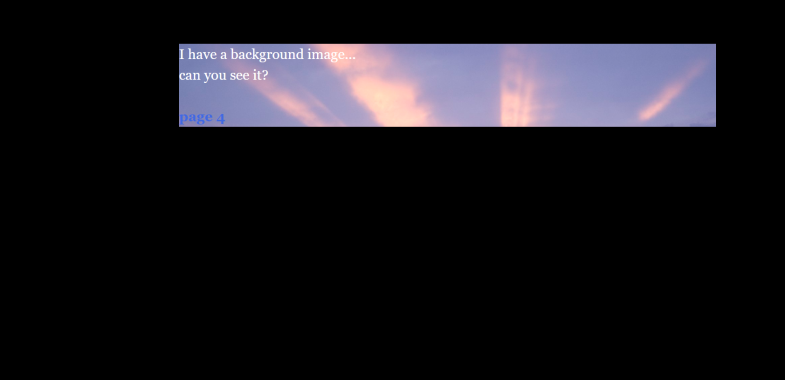


כעת, נסביר את הקוד:

**tw-story** מגדיר שמה שיגיע בסוגריים מסולסלים אחרי ישפיע על כל הסיפור, ולא רק על תוכן העמוד. תוכן העמוד נקבע לפי מה שאנו רושמים בעמוד. עד כה נתקלנו ב **tw-passage** המגדיר שמה שיגיע בסוגריים מסולסלים אחרי ישפיע **רק** על תוכן העמוד, כלומר רק על החלק המסומן באדום:



על מנת להמחיש את ההבדלים נשנה את הקוד ובמקום **tw-story** נכתוב **tw-passage**, התוצאה:

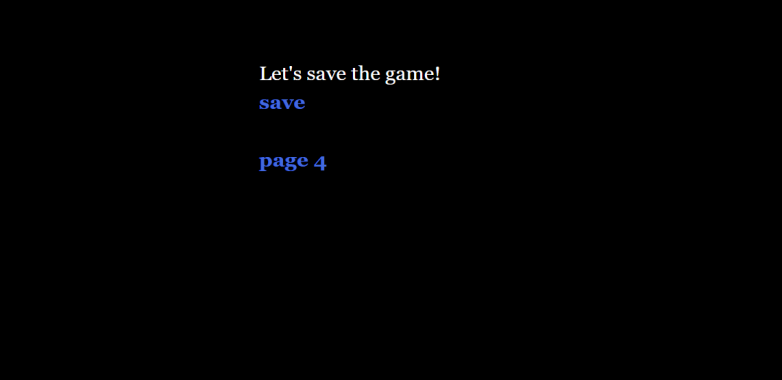


קל לראות שאכן תמונת הרקע מחסה רק את תוכן העמוד, ולא את הסיפור כולו.

**background-image: url** מקבל את כתובת הקישור של התמונה, כפי שצוין מקודם יש לוודא שאכן הקישור מסתיים ב .png או .jpg

**background-size: cover** דואג שהתמונה תכסה את כל השטח עליו היא מוגדרת.

## הוספת כפתורי שמירה וטעינה

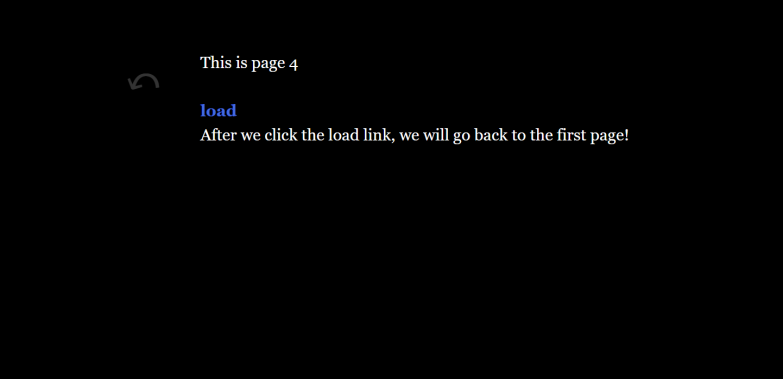
משחקי רשת רבים משתמשים בעוגיות דפדפן כדי לשמור את מיקומו של השחקן במשחק. Twine מאפשר לשמור את המשחק, כולל כל משתנים שהוגדרו והמעברים בהם ביקר השחקן. ניתן להוסיף [כפתור שמירה](https://twine2.neocities.org/#macro_save-game) השומר את מצב המשחק הנוכחי באחסון הדפדפן וכולל שם קובץ מיוחד. לאחר מכן ניתן לשחזר אותו באמצעות [כפתור טעינה](https://twine2.neocities.org/#macro_save-game).

בדוגמא ניתן לראות את הקוד של הוספת כפתור שמירה:

כעת, נסביר את הקוד:

בשורה השנייה יצרנו [קישור](https://twine2.neocities.org/#macro_link) המוצג למשתמש בעזרת המחרוזת save, לאחר שהמשתמש לוחץ על הקישור הפקודה שנעשית היא save-game, המחרוזת Slot A שבאה אחרי מגדירה את המיקום שבו ניתן לאחסן את המשחק. ניתן לאחסן במספר מקומות לא מוגבל, דרך נוספת להגדיר מיקום של שמירה היא לפי הגדרת משתנים כמו שם משתמש (נראה בהמשך) במידה ומשתמשים רבים משחקים במשחק. לאחר שהשתמש לחץ על הכפתור הוא נעלם.

בדוגמא הבאה נראה כיצד אנו ממשמים את הכפתור טעינה, לחיצה על כפתור הטעינה תסיים את המשחק הנוכחי ותחליף אותו עם משחק טעון.



בדוגמא ניתן לראות את הקוד של הוספת כפתור טעינה:

כעת, נסביר את הקוד:

בשורה השנייה יצרנו [קישור](https://twine2.neocities.org/#macro_link) המוצג למשתמש בעזרת המחרוזת load, לאחר שהמשתמש לוחץ על הקישור הפקודה שנעשית היא load-game, כלומר המשחק הנוכחי מסתיים והמשחק נטען מחדש מהנקודה שבה התבצע הsave, במקרה שלנו חוזרים לעמוד הראשון לאחר לחיצה על הקישור. המחרוזת Slot A שבאה אחרי מגדירה את המיקום שבו יש לחפש את המשחק השמור, לכן חשוב לוודא שהמחרוזת תואמת גם ב save-game וגם ב load-game.

# משתנים

[משתנים](https://twine2.neocities.org/#macro_set) משמשים לאחסון נתונים למשחק שלך. אתה יכול לאחסן ערכי נתונים תחת שמות מיוחדים לבחירתך, ולהפנות אליהם בהמשך. ישנם שני סוגים של משתנים:

1. משתנים רגילים, ששמותיהם מתחילים ב- $, נמשכים בין קטעים, ויש להשתמש בהם לאחסון נתונים הדרושים לאורך כל המשחק.
2. משתנים זמניים, ששמותיהם מתחילים ב- \_, קיימים רק בתוך העמוד או בקטע שהם מוגדרים בו {} ונשכחים לאחר הסיום. עליך להשתמש במשתנים זמניים אם אתה כותב קוד מעבר היכול להשפיע בטעות על משתנים של קטעים אחרים. זה יכול להיות חיוני בעבודה שיתופית עם מחברים אחרים העובדים על אותו סיפור באופן עצמאי, או בעת כתיבת קוד שישמש בסיפורים מרובים.

למשתנים יש מטרות רבות: מעקב אחר מה שהשחקן השיג, ניהול מצב אחר של הסיפור, אחסון סגנונות ועוד. אתה יכול להציג משתנים על ידי הצבתם בטקסט העמוד.

כעת נעבור על סוגי פונקציות, ופעולות בסיסיות שאפשר לעשות על משתנים.

למידע ודוגמאות נוספות ניתן להסתכל [כאן](https://twine2.neocities.org/#macro_set).

## הפקודה Set:

בעזרת פקודה זו ניתן להגדיר משתנים חדשים, הצורה הבסיסית שבה נשתמש על מנת להגדיר משתנה חדש או לשנות משתנה קיים היא (**set:** *variable* **to** *value*). לא ניתן להגדיר משתנים המתחילים בספרה. ניתן להגדיר משתנה לערך קבוע או להגדיר משתנה שיהיה שווה למשתנה אחר, אפשרות נוספת לי להגדיר מספר משתנים בו זמנית. דוגמאות:

1. (**set:** $*myNumber* **to** *1*).
2. (**set:** $*myVar* **to** *“1”*).
3. (**set:** *$you* **to** *“you are my number “ +* $*myVar*).
4. (**set:** *$you* **to** *“you are my number “ +* $*myNumber*). **כאן אנו נקבל שגיאה**, כי התוכנה מזהה שמדובר בשני סוגי משתנים שונים.
5. (**set**: $fruits to "Banana", $number to 6). בפקודה אחת, אנו מגדירים שני משתנים בו זמנית.
6. ((set: $array to ("B","B","B","C") בצורה זו ניתן להגדיר [מערכים](https://twine2.neocities.org/#type_array).

## הפקודה Print:

בעזרת פקודה זו המתכנת יכול להדפיס למסך את המשתנים שהוא הגדיר, כלומר אחרי ההדפסה ניתן לראות את הערך של המשתנה שהודפס גם ב Debug וגם ב Play היכן שנמצא כל התוכן של העמוד. הפקודה מתבצעת בצורה הבאה (**print**: $variable) כעת נעבור על דוגמאות נוספות:

1. (**print**: $variable+ “your string”) ניתן להדפיס משתנה ומיד אחריו מחרוזת לבחירתך.
2. )(**set**: $p **to** (**print**: "Count " + $name) ניתן לשלב בין פקודה זו לפקודה Set, כלומר בכל פעם שנרשום רק $p בתוכן העמוד אנו נדפיס את המשתנה מבלי צורך לכתוב מחדש את כל פקודת ההדפסה.

## תנאים

כמו בכל שפות התכנות שאנו מכירים, גם כאן יש אפשרות להוסיף תנאים (if, else if, else) התנאי מקבל ערך בוליאני ובמידה הוא מתקיים מבצע פקודה אחת ובמידה והוא לא מתקיים מתבצעת פקודה אחרת. אחרי כל תנאי ניתן להוסיף טקסט כלשהו בעזרת [], חשוב לשים לב שבתוכנה זו אנו כותבים [] במקום הסוגרים שאנו לרוב רגילים אליהם {}. בתוך הסוגרים ניתן להוסיף את הטקסט שיופיע בתנאי שהתנאי מתקיים, ניתן להוסיף לפני התנאים משפט כך שהתנאים ישלימו את המשפט כמו בדוגמא הבאה שנראה:

**(***set:* $goTo to 'page2'**)**

Let’s go to page **{**

**(***if:* $goTo is 'page2'**)[**

[[page 2]]

**](***else-if:* $goTo is 'page3'**)[**

[[page 3]]

**](***else:*​**)[**

[[page 4]]

leaves.

**]}**.

כעת, נסביר את הקוד מלמעלה. בשורה הראשונה אנו מגדירים את המשתנה $goTo בהתאם למה שנגדיר אותו, אחד התנאים יתקיים, לאחר מכאן בשורה השנייה אנו כותבים משפט שיופיע על המסך לפני שהפעולה שתתקיים בסוגריים. לאחר מכן מופיעים לנו שלושת התנאים שלנו, אחד התנאים יתבצע בהתאם לערך שאנו מגדירים עבור $goTo ניתן לראות שאחרי כל תנאי יש סוגריים שבתוכם יש קישור לעמוד אחר, במידה והתנאי מתקיים יופיע לנו קישור ונועבר לעמוד אחר.

במקום קישור לעמוד ניתן להוסיף גם פעולות על משתנים, כפתור שמירה/טעינה, טקסט ועוד.

## לולאות

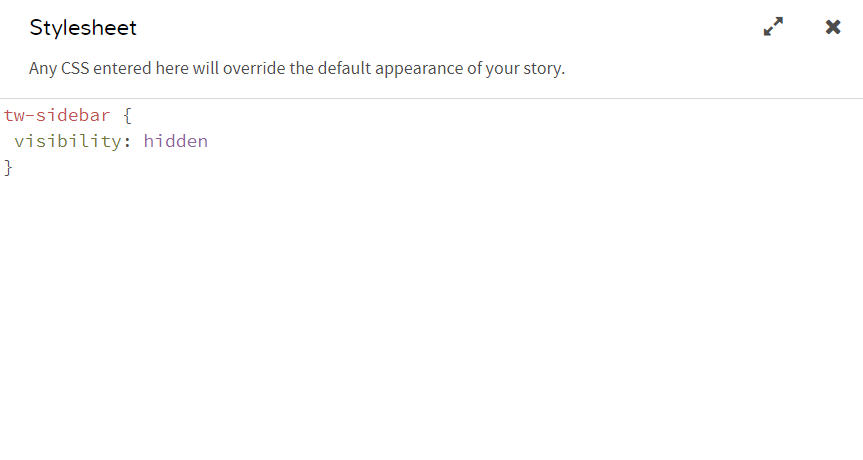
כמו בשפות תכנות רבות שאנו מכירים, גם כאן ניתן להשתמש בלולאות על מנת לבצע את אותה פעולה מספר פעמים. כל פקודה שתתבצע בעת ביצוע הלולאה יש לכתוב ב []. ישנן שתי דרכים עיקריות לכתוב לולאות:

1. (**for**: **each** $var, **...**$arr) לכל איבר במערך או בתחום מסוים (range(1,10) חשוב לכתוב את השלוש נקודות לפני.
2. (**for**: \_ingredient **where it contains** "petal", ...$reagents) מדפיס כל מחזורת במערך reagents המכילה את המילה petal.

# העלמת חץ החזרה אחורה

כאשר אנו נכנסים למצבי Debug ו- Play מופיע לנו חץ חזרה אחורה. בחלק המקרים במשחקים שלנו לא נרצה לאפשר לשחקן את האפשרות של לחזור אחורה ולשנות את בחירתו. על מנת למנוע מצב כזה ניתן להעלים ולמחוק את החץ.

על מנת להעלים את החץ נשתמש בקוד הבא:



**tw-sidebar** משפיע לנו על החץ בצד העמוד.

כאשר אנו מגדירים את ה **visibility: hidden** מגדיר לנו שהחץ יהיה מוסתר.

# ביבליוגרפיה

<https://twine2.neocities.org/>

<https://www.youtube.com/playlist?list=PLklITFhXtPCCKadv-0Gcbqoj3OCev695D>