

פונקציות Shortly – אתר Lambda

change_link_password

הסבר קצר:

fonkizia זו מעדכנת את הסיסמה של לינק ספציפי. היא בודקת בעלות על הלינק, מזودאת שהلينק موجود בסיסמה, מאמתת את הסיסמה הנוכחיות ולאחר מכן מעדכנת את הסיסמה החדשה ב-DynamoDB.

קלט לדוגמה:

```
        }
body": "{\"LinkId\": \"link123\", \"userId\": \"userA\", \"currentPassword\":"
      "\"\\\"oldPass\\\"", \"newPassword\": \"newPass123"
      {
```

delete_image

הסבר קצר:

fonkizia זו מ afsot את תמונה ה profiel של משתמש לתמונה ברירת מחדל או לתמונה placeholder ספציפית ב-DynamoDB. היא אינה מוחקת בפועל קבצים מ-S3.

קלט לדוגמה:

```
        }
body": "{\"userId\": \"user123\", \"pictureUrl\":"
      "\"\\\"https://example.com/new_default_avatar.jpg\""
```

create_mailing_list

הסבר קצר:

fonkzia zo yozret reshitat tpozcha chadsha. hiya makbelat **initiatorId** (mזהה יוצר הרשימה), shem lereshima (name), reshimat chabrim (**recipientsIds**), reshimat ctobot amayil (**recipientsEmails**). fonkzia machzira at ctobot amayil shel chabrim shzino, makhadt otton um amayilim (tor tifol bcpfiloyot), shomerat at reshima betbalat **Mailing_Lists**.

קלט לדוגמה:

```
        }
body": "{\"initiatorId\": \"userABC\", \"name\": \"My New Group\", \"recipientsIds\": \""
        ["friend1", "friend2"], \"recipientsEmails\": [\"email1@example.com\",
        \"\"email2@example.com
        {
```

get_all_links

הסבר קצר:

fonkzia zo machzira at col hlinikim mtablat **Links**. hiya mbeatzat srikha mlaahah shel htblah machzira at col hperitim shnematzao.

קלט לדוגמה (ain zorur b-body):

```
{}
```

check_unread_notifications

הסבר קצר:

fonkzia zo bokeret am lemashemtsh mosimim yishterot shudin la nakrao. hiya maamatah at kiyom mashemtsh vlaachar mkn sorkat at tbalat hterot cdi lמצוא hterot shycot lmזהה mashemtsh vottos **IsRead** hshwooh la-**0** (la nakra).

קלט לדוגמה:

```
        }
        \"body\": \"{\"UserId\": \"some-user-id-123\"
        {
```

ban_user

הסבר קצר:

פונקציה זו משנה את סטטוס הפעילות (`IsActive`) של משתמש ושל כל הלינקים שבבעלותו. היא מקבלת `UserId` ודגל בוליאני `IsActive`. אם המשתמש הופך ללא פעיל (חסום), גם הלינקים שלו מושבתים.

קלט לדוגמה:

```
        }
    "{body": "{\"UserId\": \"userXYZ\", \"IsActive\": false"
    {
```

get_all_active_links

הסבר קצר:

פונקציה זו ממחזירה את כל הלינקים בהם גם ציבוריים וגם פעילים. היא מבצעת סריקה של טבלת ה-`Links` ומסנןת את התוצאות כדי לכלול רק לינקים שבהם `IsActive` הוא `true`.

קלט לדוגמה (אין צורך ב-body):

```
{}
```

delete_link

הסבר קצר:

פונקציה זו "מבטלת" לינק על ידי שינוי סטטוס `IsActive` שלו ל-`false` ב-`DynamoDB`. היא אינה מוחקת את הلينק בפועל מהטבלה, אלא מסמנת אותו כלא פעיל.

קלט לדוגמה:

```
        }
    "{\"body\": {\"LinkId\": \"link-to-deactivate-123"
    {
```

get_all_notifications

הסבר קצר:

פונקציה זו מחזירה את כל התראות עבור משתמש נתון, וממיינת אותן לבקשת חברות ממתינות והתראות אחרות (שאין נקרו ובעלותן 7 ימים אחרונים). עבור בקשות חברות, היא גם מחזירה את פרטי השולח (שם משתמש ותמונה) מטבלת המשתמשים.

קלט לדוגמה:

```
        }
    "body": "{\"userId\": \"some-user-id-123\"
        {
```

get_user_mailing_lists

הסבר קצר:

פונקציה זו מחזירה את כל רשימות התפוצה שנוצרו על ידי משתמש מסוים ([InitiatorId](#)). היא מבצעת פעולה [Scan](#) על טבלת [Mailing_Lists](#)-ב-[DynamoDB](#), כולמר סורקת את כל הطلبיה, ומסננת את התוצאות לפי זהה המשתמש שסופק. הפונקציה מחזירה רק את מזהה הרשימה ([ListId](#)) ושם הרשימה ([ListName](#)).

קלט לדוגמה (מה שהפונקציה מקבלת):

```
{"body": {"InitiatorId": "user-123-abc"}
```

get_link_details

הסבר קצר:

פונקציה זו מספקת מידע מפורט על קישור אחד. היא שולפת את הגדרות הקישור (כמו האם הוא פרט או מוגן בסיסמה), סופרת את סך הלחיצות עלי, ומסכמת את נתוני הלחיצות לפי מדינות. המידע נאוסף מטבלת [Links](#).

קלט לדוגמה:

```
{ "body": {"linkId": "link-xyz-789"}}
```

new_notif

הסבר קצר:

פונקציה זו יוצרת התראה חדשה ושומרת אותה במסד הנתונים (טבלת **Notifications**). היא מקבלת את פרטי התראה (למי מיעדת, התוכן, וכו'), מייצרת עבורה מזהה "יחודי" (**NotifId**), ומוסיפה אותה לטבלה.

קלט לדוגמה:

```
{
  "body": "{\"ToUserId\": \"user-456\", \"FromUserId\": \"user-123\", \"Text\": \"You have\n      a new friend request!\", \"LinkId\": \"link-abc\"}"
}
```

get_links_by_user_id

הסבר קצר:

פונקציה זו מחזירה את כל הקישורים שנוצרו על ידי משתמש ספציפי. היא מנסה לבצע שאילתת עילה באמצעות אינדקס (**GSI**), ואם האינדקס לא קיים, היא חוזרת לחולפה פחות עילה של סריקה כל הטבלה כדי למצוא את הקישורים הרלוונטיים.

קלט לדוגמה:

```
{
  "body": "{\"UserId\": \"user-xyz-123\"}"
}
```

get_user_links

הסבר קצר:

פונקציה זו מחזירה את כל הקישורים הציבוריים (**IsPrivate: false**) והפעילים (**IsActive: true**) של משתמש מסוים. היא מבצעת שאילתת כדי לקבל את כל הקישורים של המשתמש, ולאחר מכן מסננת אותם ברמת הקוד כדי להציג רק את אלו שעונים על שני התנאים.

קלט לדוגמה:

```
{
  "body": "{\"UserId\": \"a1\"}"
}
```

get_user_by_id

הסבר קצר:

פונקציה זו מרכיבה פרופיל משתמש מלא על ידי איסוף נתונים ממספר טבלאות. היא שולפת מידע בסיסי על המשתמש, את רשימת ההישגים שלו (כולל פרטי ההישג), ואת הקישורים שיצר. הפונקציה גם מבצעת סינון פרטיות: אם המשתמש שצופה בפרופיל אינו בעל פרופיל, קישורים פרטיים יוסתרו מההתשובה.

קלט לדוגמה:

```
{ "body": "{\"ProfileOwnerId\": \"user-profile-456\", \"LoggedInUserId\": \"user-viewer-123\"}" }
```

is_user_banned

הסבר קצר:

פונקציה זו בודקת אם חשבון משתמש מסוים פעיל או חסום. היא ניגשת לטלית המשתמשים, שולפת רק את השדה **IsActive** (כדי לחסוך במשאבים), ומחזירה תשובה בוליאנית (**true** אם פעיל, **false** אם לא).

קלט לדוגמה:

```
{
  "body": "{\"userId\": \"user-to-check-789\"}"
}
```

mark_notifications_as_read

הסבר קצר:

פונקציה זו מסמנת את כל התראות של משתמש מסוים כ"נוקראו". היא מattaרת את כל התראות שטרם נקרו או עברו המשתמש באמצעות שאילתה, ולאחר מכן עוברת על כל אחת מהן וمعدכנת את הסטטוס שלה ל"נוקרא" (**IsRead = 1**).).

קלט לדוגמה:

```
{
  "body": "{\"userId\": \"user-abc-456\"}"
}
```

get_public_links

הסבר קצר:

פונקציה זו מיועדת להציגת "פיד" ציבורי של קישורים. היא סורקת את כל טבלת הקישורים ומסננת אותם ברמת מסד הנתונים כדי לקבל רק קישורים שהם גם ציבוריים (`IsPrivate: false`) וגם פעילים (`IsActive: true`). היא גם מבקשת רק שדות מידע ספציפיים כדי ליעיל את השילפה.

קלט לדוגמה (אין צורך ב-body):

```
{}
```

get_user_friends

הסבר קצר:

פונקציה זו מחזירה את רשימת החברים של משתמש מסוים. היא קוראת את רשימת מזהה החברים מהפרופיל של המשתמש, אז עבור כל מזהה, היא פונה שוב למסד הנתונים כדי לשולף את הפרטים הבסיסיים של אותו חבר (שם, תמונה, אוימיל).

קלט לדוגמה:

```
{
  "body": "{\"UserId\": \"main-user-123\"}"
}
```

respond_to_friend_request

הסבר קצר:

פונקציה זו מנהלת את המענה לביקשת חברות. היא מקבלת את מזהה ההתראה של הבקשה (`NotifId`) ואת החלטת המשתמש (אישור/דחייה). היא מעדכנת את סטוס הבקשה המקורי, שלוחת התראה חדשה לשולח המקורי עם התשובה, ואם הבקשה אושרה, היא מעדכנת את רשימות החברים (`Friends`) של שני המשתמשים.

קלט לדוגמה:

```
{
  "body": "{\"NotifId\": \"notif-abc-123\", \"accept\": true}"
}
```

restore_link

הסבר קצר:

פונקציה זו משחררת קישור שנמחק (מחיקה "רכה") על ידי הגדלת הדגל **IsActive** שלו בחזרה ל-**true**. היא מקבלת מזהה קישור, ומשתמשת בפועלות עדכון עם תנאי (**ConditionExpression**) כדי לוודא שה קישור קיים לפני ביצוע השחזר.

קלט לדוגמה:

```
{  
  "body": "{\"LinkId\": \"link-to-restore-123\"}"  
}
```

set_link_password

הסבר קצר:

פונקציה זו מגדירה סיסמה ל קישור. היא דורשת מזהה קישור, מזהה משתמש וסיסמה חדשה. הפעולה תצליח רק אם המשתמש הוא הבעלים של הקישור וה קישור אינו מוגן בסיסמה קיימת **attribute_not_exists(IsPasswordProtected) OR IsPasswordProtected = :p_false**. זה מונע דרישת של סיסמות קיימות ומבטיח שרק הבעלים יכול להגדיר סיסמה.

קלט לדוגמה:

```
{  
  "body": "{\"LinkId\": \"link-unsecure-789\", \"userId\": \"owner-user-123\",  
           \"newPassword\": \"MySecret123\"}"  
}
```

post_create_user

הסבר קצר:

פונקציה זו מופעלת באופן אוטומטי על ידי שירות AWS Cognito מיד לאחר שימוש חדש מס'ם הרשמה. היא לוקחת את פרטי המשתמש מהירוע של Cognito, מכינה רשומה פרופיל חדשה עם ערכי ברירת מחדל (כמו תאריך הצטרפות, סטטוס פעיל וכו'), ושומרת אותה בטבלת המשתמשים (Users) ב-DynamoDB. זו הדרך לסנן משתמשים חדשים מערכות האימות למסד הנתונים של האפליקציה.

קלט לדוגמה (ירוע מ-Cognito):

```
{  
    "request": {  
        "userAttributes": {  
            "sub": "cognito-unique-id-123",  
            "email": "test@example.com",  
            "name": "Test User",  
            "nickname": "tester"  
        }  
    }  
}
```

new_short_url

הסבר קצר:

פונקציה זו יוצרת קישור מקוצר חדש. היא מקבלת כתובת URL ארוכה ומזהה משתמש, מייצרת קוד קצר ייחודי, ובודקת שאין קיים כבר בטבלה. היא שומרת את הקישור החדש בטבלת Links, ובנוסף, מעדכנת את רשומה המשתמש בטבלת Users על ידי הוספה מזהה הקישור החדש לרשימה הקישורים המשויכת אליו.

קלט לדוגמה:

```
{  
"body": {"\"url\": \"https://www.wikipedia.org/wiki/Very_Long_Article_Name\",  
        \"userId\": \"user-abcd\""}  
}
```

remove_link_password

הסבר קצר:

פונקציה זו מסירה הגנת סיסמה מ קישור. היא מקבלת מזהה קישור ומזהה משתמש. הפונקציה מבצעת עדכון בטבלת הקישורים, אך ורק בתנאי (ConditionExpression) שהמשתמש הוא הבעלים של הקישור ושה קישור אכן מוגן כרגע בסיסמה (`IsPasswordProtected = :p_true`). אם התנאים מתקיימים, ההגנה והסיסמה מוסרות.

קלט לדוגמה:

```
{  
"body": "{\"linkId\": \"link-secure-456\", \"userId\": \"owner-user-789\"}"  
}
```

send_friend_request

הסבר קצר:

פונקציה זו שולחת בקשה חברות משתמש אחד לאחרר. היא מבצעת מספר בדיקות חיוניות: שהמשתמשים קיימים, שהם לא אותו אדם, ושאין ביניהם כבר בקשה חברות קיימת (מכל סוג). אם כל הבדיקות עברו, היא יוצרת התראה חדשה מסוג "pending" המיצגת את בקשה החברות.

קלט לדוגמה:

```
{  
"body": "{\"FromUserId\": \"user123\", \"ToUserId\": \"user456\"}"  
}
```

request_image_upload_url

הסבר קצר:

פונקציה זו מייצרת כתובת URL מאובטחת וזמןית (Pre-signed URL) המאפשרת לשלוח צד- לקוח להעלות קובץ שירות לאחסון S3. היא מקבלת את סוג התוכן של הקובץ (`contentType`), יוצרת שם קובץ ייחודי, ומחזירה את כתובת-URL להעלאת (`uploadUrl`) ואת כתובת הציבורית הסופית של הקובץ (`finalUrl`) לאחר שההעלאה תושלם.

קלט לדוגמה:

```
{  
"body": "{\"contentType\": \"image/png\"}"  
}
```

send_mail

הסבר קצר:

fonkzia'a zo sholchot aimil shi'tof ubor kishor l'reshimot nenuim. hia aosft ctobot aimil mmasfer makorot (r'shimot t'faza, chirim, ctobot shahozno idnit), yozrat r'shima "y'hodit", v'mcina aimil motam ai'shit. libsof, hia matchbarat l'shurat dor (SMTP) batzutim prutim shmorim b'mashani s'biba v'sholchot at ha'aimil l'kul nenu.

kul't l'dogma:

```
{
  "body": "{\"senderId\": \"user-A\", \"linkId\": \"link-123\", \"groupIds\":\n    [\"group-alpha\"], \"friendIds\": [\"user-B\"], \"recipientsEmails\":\n    [\"new@example.com\"], \"site\": \"https://shortly.com/link-123\"}"
}
```

get_all_users_with_stats

הסבר קצר:

fonkzia'a zo machzira at kl' m'shatmashim m'msd ha'ntoinim v'mospekha midu stutischi' ul'ihem. am mospekha user Id b'kul't, hia gm machzira at halinkim m'shatmashim la'otnu m'shatmash.

kul't l'dogma:

```
{
  "body": "{\"userId\": \"894980c8-e8a6-4921-9bb0-f917671caa65\""
}
```

toggle_link_privacy

הסבר קצר:

fonkzia'a zo m'shena at stutios ha'prutiyot (IsPrivate) shel link ntuon. hia m'kiblat link Id v'koveuta at m'zbu ha'nocchi' shel halink. la'achr m'cn, hia hovcat at stutios (m'pruti l'zibori au m'zibori l'pruti) v'mudcna at halink b'msd ha'ntoinim.

kul't l'dogma:

```

  }
  {
    "\"body\": \"{\"linkId\": \"someLinkId123\""
  }
```

verify_link_password

הסבר קצר:

פונקציה זו בודקת סיסמה עבור 링ק מסוים. היא מקבלת **linkId** וסיסמה. אם הリンク קיים והוא מוגן בסיסמה, הפונקציה משווה את הסיסמה שהזונה עם הסיסמה המאוחסנת. אם הן תואמות או אם הリンク אינו מוגן בסיסמה כלל, היא מעניקה גישה.

קלט לדוגמה:

```
        }
    {"body": {"LinkId": "someLinkId123", "password": "mysecretpassword"
        {
```

verify_admin_status

הסבר קצר:

פונקציה זו בודקת אם משתמש הוא חלק מקבוצת "Admins" במאגר משתמשים של **Cognito**. היא קובעת באופן דינמי את זהה מאגר המשתמשים מהתוור שם עירימת **CloudFormation** ואת שם הקבוצה מסביבה הרצויה.

קלט לדוגמה:

```
        }
    {"body": {"UserId": "username@example.com"
        {
```

update_user_profile

הסבר קצר:

פונקציה זו מעדכנת את פרטי הפרופיל של משתמש במסד הנתונים של **DynamoDB**. היא מקבלת **userId** ושמות נוספים נוספים כמו **FullName**, **Country**, **Picture**. היא בונה ביטוי עדכון דינמי ומעדכנת רק את השדות שסופקו בקלט.

קלט לדוגמה:

```
        }
    body": {"userId": "user123", "FullName": "Jane Doe", "Country": "
        {
            "Country": "Canada"
        }
```

track_click

הסבר קצר:

פונקציה זו מטפלת בUrls של לינקים. היא מחזירה את פרטי הלינק, כולל אם הוא פעיל. אם הקLIK אין של בעל הלינק עצמו, היא מעלה את מונה הקLIKים של הלינק. היא גם בודקת אם הקLIK מגע לאבני דרך המזוכות בהישגים, ובהתאם לכך מעניקה הישגים, שולחת התראות וمعدכנת את רשומות המשתמש. בנוסף, הפונקציה בודקת אם הלינק מוגן בסיסמה; אם כן, היא מחזירה דגל המציין זאת במקום את כתובות ה-[URL](#)-המקורית.

קלט לדוגמה:

```
        }
    "{\"body\": {\"code\": \"test-link-123\", \"userId\": \"test-user-456\"}
        {
```

get_all_active_users

הסבר קצר:

פונקציה זומחזירה את כל המשתמשים הפעילים (שבהם [IsActive](#) מוגדר ל-[true](#)) מטבלת ה-[Users](#) ב-[DynamoDB](#). היא מבצעת סריקה של הטבלה ומסננת את התוצאות.

קלט לדוגמה (אין צורך ב-body):

```
{}
```