



Full Stack- server- PHP

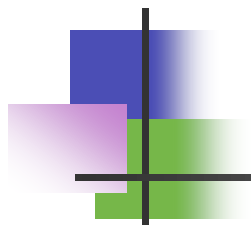
הרצאה 1

מבוא

סביבת עבודה

קוד PHP בסיסי

מטרת הקורס



- חשיפה לשפת PHP
- פתרון בעיות בסביבה זו
- שימוש במבני נתונים הכלולים בשפה
- בניית מערכות מכוונות עצמים המארחות
בסיסי נתונים רלציוניים

סביבת עבודה

■ תוכנות להתקנה

■ Xampp – שרת וירטואלי המדמה שרת חיצוני apache מבוסס מערכת הפעלה UNIX

■ קישור להורדה:

<https://www.apachefriends.org/index.html>

■ Notepad+ - תוכנת עריכת טקסטים

■ קישור להורדה:

<https://notepad-plus-plus.org/>

■ Google Chrome - תצוגת תוכן אתרים

■ קישור להורדה:

<https://www.google.com/chrome/>



Xampp

- Xampp – שרת וירטואלי המדמה שרת חיצוני apache מבוסס מערכת הפעלה UNIX קישור להורדה: ■

<https://www.apachefriends.org/index.html>

- לאחר ההתקנה ניתן להשתמש ב- MySQL
- נפעיל את xampp control panel
- נפעיל באמצעות לחיצת start את:
 - Apache
 - MySQL

- מעבר לעמוד הראשי של Xampp



תקלה בהרצת Apache

Error: Apache shutdown unexpectedly

■ לחצו על httpd.conf -> config

■ שנו את השורות הבאות:

■ Listen 80 -> Listen 8080

■ ServerName localhost:80->ServerName localhost:8080

■ שמירה ויציאה מהקובץ

■ לחצו על httpd-ssl.conf -> config

■ שנו את השורות הבאות:

■ Listen 443->Listen 4433

■ <VirtualHost _default_:443>-><VirtualHost _default_:4433>

■ ServerName localhost:443->ServerName localhost:4433

phpMyAdmin



Welcome to XAMPP for Windows 8.0.10

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but that are insecure if you want to have your installation accessible to others. If you want have your XAMPP accessible from the internet, make sure you understand the implications and you checked the [FAQs](#) to learn how to protect your site. Alternatively you can use [WAMP](#), [MAMP](#) or [LAMP](#) which are similar packages which are more suitable for production.

Start the XAMPP Control Panel to check the server status.

Community

XAMPP has been around for more than 10 years. There is a large community behind it. You can ask for help or help others on the [Community](#) page.

■ תקשורת ל-DB

■ עבור Port 80

<http://localhost>

■ עבור port 8080

<http://localhost:8080/>

כניסה בסרגל כלים על
phpMyAdmin



- דף אינטרנט מורכב מ:

- HTML

- JavaScript

- תמונות ורכיבים נוספים (אודיו, וידאו וכו')

- על מנת ליצור מספר רב של דפים, התוכנה מייצרת מחדש דף HTML אליו הוא רוצה להכנס.

- יצירת הדפים הינה דינמית, אתרים מבוססי PHP נקראים אתרים דינמיים.



צד שרת/צד לקוח-client /server

צד הלקוח מבחינת התוכנה הוא כל מה שקורה על המסך של המשתמש, דרך הדפדפן בו הוא גולש או ה"חזית" של האפליקציה.

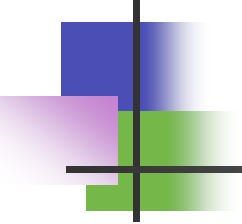
צד השרת הוא מחשב היושב בדרך כלל במקום מרוחק ו"מגיש" את הנתונים הנדרשים לדפדפן. למשל, אם בפייסבוק תלחצו על כפתור כדי להציג את רשימת החברים שלכם, הלקוח ישלח את הבקשה לשרת כי בצד השרת שמורים כל הנתונים (כמו גם רשימת החברים). השרת יחזיר את רשימת החברים שיוצגו על המסך, כאשר לרוב צד הלקוח יקבע איך הם יוצגו.

בשנים האחרונות צד הלקוח הולך ונהייה יותר ויותר עוצמתי ביכולותיו ועושה דברים שבעבר היו נחלת השרת בלבד.

טכנולוגיות נפוצות בצד לקוח הן: HTML, CSS, JavaScript, Angular, React

טכנולוגיות נפוצות בצד שרת: PHP, C, C#, .NET, Python, Node.js

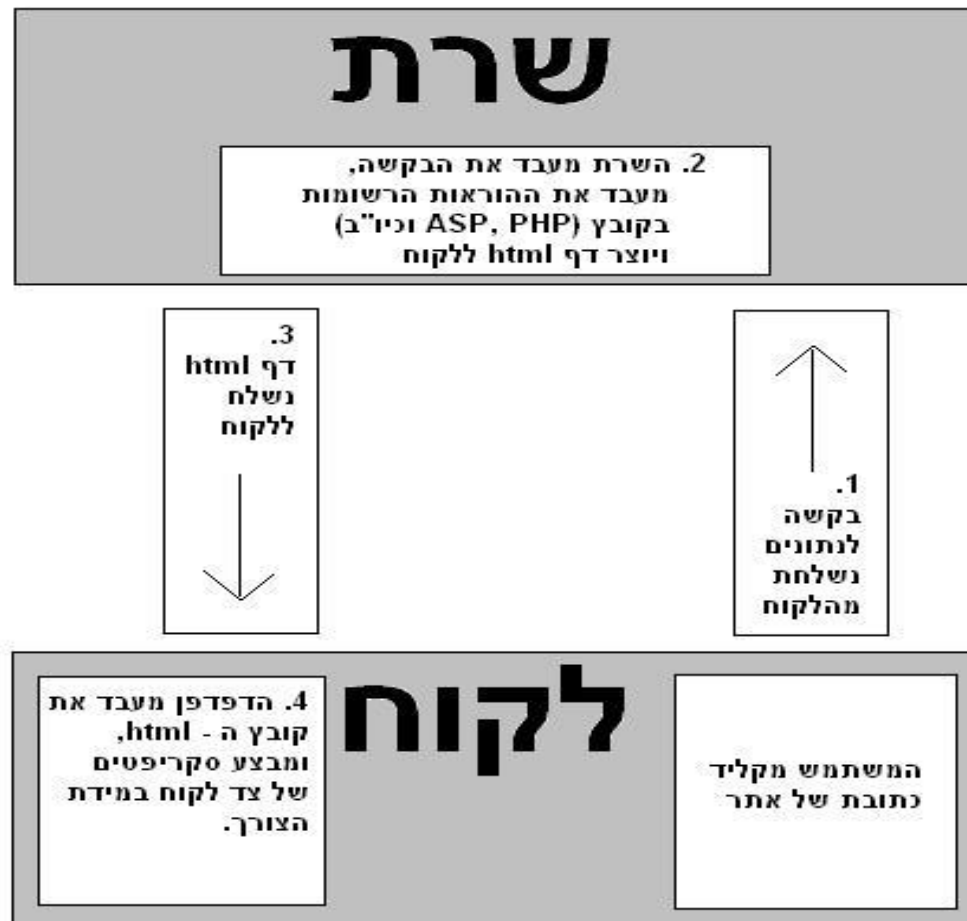
צד שרת וצד לקוח – איך זה עובד?



קוד צד לקוח הוא קוד HTML סטטי. למעשה הוא רק מציג את התוכן בצורה מעוצבת ללא כל חישוב, קריאה לבסיסי נתונים, גישה לקבצים, וכו'..

קוד צד שרת הוא קוד שמבצע עיבוד על השרת-גישה לבסיס הנתונים, חישובים מתמטיים, גישה לקבצים וכן הלאה. ולאחר כל העיבוד הוא מייצר קוד צד לקוח ושולח אותו לדפדפן. הדפדפן רק מציג את התוצר של העיבוד שנעשה על השרת.

צד שרת וצד לקוח





הקדמה – שפת PHP

- **שפת PHP** היא שפת סקריפט שרצה בצד השרת <= יש צורך להעלות דף PHP לשרת
- דף אינטרנט מורכב בעיקר מ-HTML, תמונות, JavaScript ורכיבים נוספים (כמו קבצי וידאו ואודיו)
- שפת PHP מאפשרת יצירת דפי HTML בצורה דינאמית
 - כאשר אנו גולשים לאתר סטטי, אנו מקלידים את כתובת האתר בדפדפן ו'נכנסים' לאתר, האתר מחזיר לנו קובץ HTML שמוצג על הדפדפן.
 - כאשר אנו גולשים לאתר דינמי, אנו מקלידים את כתובת האתר בדפדפן ו'נכנסים' לאתר, האתר מריץ קוד PHP שיוצר את קובץ ה-HTML והוא מוצג על הדפדפן.

אתר סטטי לעומת אתר דינמי

הקלדת כתובת האתר
בדפדפן
כניסה לאתר
האתר מחזיר לנו קובץ
HTML



הקלדת כתובת האתר
בדפדפן
כניסה לאתר
האתר מריץ קוד PHP שיוצר
את קובץ ה-HTML





שפת PHP

- שפת סקריפט פופולרית (ויקיפדיה, פייסבוק..) שרצה בצד השרת (מופעלת ע"י שרת אינטרנט)

- PHP: Hypertext Preprocessor

- קבצי php יכולים להכיל:

- Text
- HTML
- CSS
- JavaScript
- PHP

- קוד PHP רץ על השרת והתוצאה חוזרת כ-HTML



מה PHP יכול לעשות

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data



יתרונות PHP

- רץ במגוון פלטפורמות

(Windows, Linux, Unix, Mac OS X, etc.)

- תואם כמעט את כל השרתים נכון להיום

(Apache, IIS, etc.)

- תומך במגוון רחב ש DB

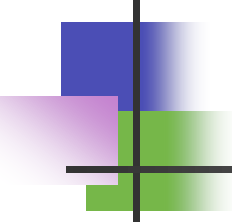
- חינמי

- קל ללמידה

שימוש ב-PHP

- PHP רצה על שרת האינטרנט (ולכן לא ניתן לפתוח קובץ PHP על הדפדפן)
- על מנת להריץ קוד ב-PHP יש להתקין PHP על המחשב.
- כאשר מעלים קוד PHP לשרת האינטרנט, יש לבדוק שמותקן עליו PHP
- ניתן להתקין PHP על מערכת הפעלה Windows או מבוססת Linux

כתיבת קוד



- כאשר נכתוב קוד PHP ב- notepad+ נשמור את הקובץ עם סיומת php.

- ניתן לכתוב קוד PHP בדף HTML או כקוד php ללא.

- תחילת קוד PHP תתחיל בצורה הבאה:

`<?php`

על מנת לסמן לשרת שמדובר בקוד php

- כאשר הקוד משולב עם HTML נקפיד לסגור את תגית ה-PHP ע"י `>?`



כללים בכתיבת קוד

- פקודת Print מבצעת הדפסה על המסך של כל מה שמופיע אחריה
 - טקסט מוקף במרכאות 'text'
 - בסוף פקודה יהיה נקודה פסיק ;
 - כתיבת פקודות, תנאים, מחלקות, פונקציות וכדו' אינן רגישות לאותיות רישיות (קטנות/גדולות)
- (if, else, while, echo)



בניית דף PHP - דוגמא

- ניתן לשלב קוד PHP בתוך דף HTML וניתן לכתוב קוד PHP נקי
- להלן דוגמא לקוד עבור דף המדפיס את הטקסט "Hello World":

```
<?php  
    print "Hello World!";  
?>
```

■ הסבר הקוד:

- ראשית יש לנו את תגית `<?php` שהיא קריטית על מנת 'לסמן' לשרת שמדובר פה בקוד PHP ולא במשהו אחר - ללא התגית הזו שום קובץ PHP לא יעבוד.
- אחרי כן אנו משתמשים בפקודת `print` שמדפיסה אל המסך כל מה שבא אחריה – במקרה הזה טקסט.
 - הטקסט צריך להיות מוקף במרכאות.
 - נקודה פסיק (;) מסמנת את סוף הפקודה.
- נשמור את הדף עם סיומת `php` וננסה להריץ



הרצת דף PHP

- ניקח את הדוגמא לקוד עבור דף המדפיס את הטקסט "Hello World":

```
<?php  
    print "Hello World!";  
?>
```

- כעת נשמור את הדף על השרת ונריץ דרכו מה תהיה התוצאה?



קוד ראשון

■ צרו תיקיה labs ובתוכה תיקיה בשם lab1 תחת:

C:\xampp\htdocs

■ צרו קובץ firstPhp.php בתיקיית:

C:\xampp\htdocs\lab1

בתוך הקובץ כתבו:

```
<?php  
print "First PHP file"  
?>
```

שמרו את הקובץ והריצו ע"י:

<http://localhost:8080/labs/lab1/firstPhp.php>



הדפסה

- הדפסה מבוצעת ע"י 2 פקודות עיקריות

- Echo

- אין ערך מוחזר

- יכול לקבל מספר פרמטרים `echo()`

- Print

- יש ערך מוחזר (1)

- יכול לקבל ארגומנט אחד `print()`

- פקודת `echo` מהירה יותר מפקודת `print`



בניית דף PHP

- להלן דוגמא לקוד עבור דף המדפיס את הטקסט "Hello World" ע"י

```
<?php  
    echo "Hello World!";  
?>
```

- נשים לב שהרצת הדף בשרת תיתן לנו את אותה תוצאה, אם כך מה ההבדל בין הפקודה print ל-echo?
 - print הינה פונקציה שמחזירה ערך, echo איננה פונקציה אלא מילה שמורה.
 - בפונקציה print ניתן להשתמש בביטויים מורכבים אולם ב-echo לא ניתן לבצע זאת
 - echo מהיר יותר מ-print (למרות שבד"כ זה איננו מורגש)



ירידת שורה

■ מתבצעת באמצעות התווית `br`

```
echo "<br>";
```

```
print "<br>";
```




הדפסה באמצעות echo

```
<?php
echo "<h1>echo test!<h1>";
echo "echo test!<br>";
echo "echo test!<br>";
echo "echo", "test", "!";
?>
```

פלט:

```
echo test!
echo test!
echo test!
echo test!
```



הדפסת משתנים באמצעות echo

```
<?php
```

```
$txt1 = "echo test";
```

```
$txt2 = "PHP";
```

```
$x = 5;
```

```
$y = 4;
```

```
echo "<h2>" . $txt1 . "</h2>";
```

```
echo $txt2."<br>";
```

```
echo $x + $y;
```

```
echo "<br>";
```

```
Echo "$x + $y";
```

```
?>
```

פלט:

echo text

PHP

9

5 + 4



הדפסת באמצעות print

```
<?php
$num1= 5;
$num2= 2;
print "<h2>print test!</h2>";
print "print test!<br>";
print "print test!";
print "<br>".$num1."<br>";
print $num1+$num2;
print "<br>";
print "$num1+$num2<br>";
print '$num1+$num2';
```

```
?>
```

print test!

print test!

print test!

5

7

5+2

\$num1+\$num2



הערות בקוד PHP

- לפעמים אנו נדרשים להכניס הערות בטקסט, זה יכול להיות איזשהו פירוט בשבילנו או עבור מתכנתים אחרים הצריכים לתחזק את הקוד.
- קוד טוב הוא גם קוד מתועד המפשט את הבנתו למקרה ואנחנו (או מישהו אחר) ירצה להבין אותו בעתיד.
- הערה זוכה להתעלמות מוחלטת על ידי PHP כך שאנחנו יכולים להכניס כל מה שעולה בדעתנו.
- הערה מסומנת כ:
 - // עבור הערה של שורה אחת
 - /* בתחילת ההערה ו- */ בסופה עבור הערה בעלת מספר שורות



הערות בקוד PHP - דוגמאות

```
<?php
    //This is one line remark. PHP will
    ignore this line
```

■ הערה עם שורה אחת:

```
<?php
    print "Hello World!"; //message to
    show
```

■ הערה עם שורה אחת
בסוף שורת קוד:

```
<?php
/* This is
Multi
Line
Remark
*/
?>
```

■ הערה עם מספר
שורות:



משתנים בקוד PHP

- ב-PHP כמו בכל שפה, יש לנו משתנים.
- **משתנים ב-PHP** הם סוג של עצמים שלתוכם אנחנו יכולים להכניס ערכים שונים לשימושים כלשהם.
- לדוגמא: ניצור משתנה ששמו הוא `x` ונכניס לתוכו ערך מסוים ואז נדפיס אותו אל המסך באמצעות `print`

```
<?php
    $x = 5;
    print $x;
?>
```



שמות של משתנים

- כל משתנה ב-PHP מתחיל בסימן דולר (\$) ואז שם כלשהו
- משתנה חייב להתחיל באות אנגלית או קו תחתון '_'
- במידה ושם המשתנה מורכב משתי מילים או יותר, יש להשתמש בקו תחתון להפרדה ביניהן
- אנחנו יכולים להכניס למשתנה לא רק מספרים אלא גם טקסט.
- כל מחרוזת טקסט חייבת להיות מוקפת במרכאות:

```
<?php
    $my_var = "Hello, I am Text String";
    print $my_var;
?>
```



משתנים – תרגיל 1

■ משתנה כשמו כן הוא, והוא יכול להשתנות

■ מה יודפס?

```
<?php
    $my_var = 5;
    $my_var = 10;
    print $my_var;
?>
```

ניתן להכניס למשתנה גם מספרים לא שלמים, כמו למשל:

```
<?php
    $my_var = 3.14;
    print $my_var;
?>
```




משתנים מסוג מחרוזת

- אם יש משתנה שיש בו מחרוזת טקסט, ניתן לחבר ביניהם באמצעות נקודה (.)

```
<?php
    $my_var = "World";
    $another_var = "Hello ".$my_var;
    print $another_var;
?>
```

- על מנת לחבר בין שתי מחרוזות טקסט, ניתן להשתמש במרכאות כפולות
- כאשר שמים משתנה בתוך מרכאות כפולות מקבלים את מחרוזת הטקסט בתוכו

```
<?php
    $my_var = 'World';
    $another_var = "Hello $my_var";
    print $another_var;
?>
```



משתנים – תרגיל 2

■ מה יודפס?

```
<?php  
  $my_var = 10;  
  $another_var = "Hello $my_var";  
  print $another_var;  
?>
```



פעולות מתמטיות - חיבור

■ חיבור נעשה באמצעות הסימן +

```
<?php
    $my_var = 2;
    $another_var = 3;
    $result = $my_var + $another_var;
    print $result;
?>
```

- ראשית אנו מכניסים את הערך 2 לתוך משתנה `my_var` ואחרי כן את הערך 3 ל- `another_var` בסופו של דבר, הפעולה המתמטית נעשית ומושמת לתוך משתנה בשם `result` שהוא מודפס בתורו.
- התוצאה כמובן היא 5.



פעולות מתמטיות - חיסור

■ חיסור נעשה באמצעות הסימן -

```
<?php
    $my_var = 2;
    $another_var = 3;
    $result = $my_var - $another_var;
    print $result;
?>
```

■ התוצאה כמובן היא -1



פעולות מתמטיות - כפל

■ כפל נעשה באמצעות הסימן *

```
<?php
    $my_var = 2;
    $another_var = 3;
    $result = $my_var * $another_var;
    print $result;
?>
```

■ התוצאה היא ...



פעולות מתמטיות - חילוק

■ חילוק נעשה באמצעות הסימן /

```
<?php
    $my_var = 2;
    $another_var = 3;
    $result = $my_var / $another_var;
    print $result;
?>
```

■ התוצאה תהיה $2/3$ או יותר נכון: 0.666666666666667



פעולות מתמטיות - שארית

- אם מחלקים מספר במספר והתוצאה היא לא שלמה, מה שנותר הוא השארית
- שארית נעשה באמצעות הסימן %

```
<?php
    $my_var = 9;
    $another_var = 7;
    $result = $my_var % $another_var;
    print $result;
?>
```

■ התוצאה תהיה ...



חיבור וחיסור מקוצרים

- אם יש מספר ורוצים להוסיף לו את הערך 1, ניתן להשתמש בחיבור מקוצר עם הסימן ++

```
<?php
    $my_var = 5;
    $my_var++;
    print $my_var;
?>
```

- ניתן להשתמש גם בחיסור מקוצר עם הסימן --

```
<?php
    $my_var = 5;
    $my_var--;
    print $my_var;
```




בדיקות על משתנים

- ראינו שבניגוד לשפת C למשתנה איננו מגדירים סוג
- אם כן, כיצד נדע מה סוג המשתנה נקלט לפני שנבצע עבורו בדיקות או פעולות? כיצד נדע האם המשתנה מוגדר (מאותחל)?
- כדי לבדוק את סוג המשתנה וערכו ניתן להשתמש בפונקציה `var_dump()`

```
<?php
    $x = 10.365;
    var_dump($x);
?>
```

- כדי לבדוק את סוג המשתנה בלבד ניתן להשתמש בפונקציה `gettype()`

```
<?php
    $x = 10.365;
    echo gettype($x);
?>
```

- 
- כדי לבדוק אם משתנה הוא מסוג מסויים ניתן להשתמש בפונקציות `is_<type>()`

למשל: `is_int()` או `is_string()` וכדומה

```
<?php
$a = 32;
echo "a is "
is_int($a)
?>
```

יודפס 1 a

- הפונקציה `isset()` מחזירה אמת אם המשתנה מוגדר, ושקר אם המשתנה לא מוגדר (לא קיים ו/או מכיל ערך "NULL")

```
<?php
    $a = 0;
    echo "Variable a
is ". isset($a);
?>
```



תרגול

https://download.cnet.com/WampServer-64-Bit/3000-10248_4-75544590.html

1. כתוב תכנית אשר מבצעת השמה למשתנה (מספר כלשהו) והתכנית מדפיסה למסך האם המספר הוא זוגי או אי-זוגי (even/not even). ההנחה היא שמדובר במספרים חיוביים בלבד.
2. כתוב תכנית אשר מבצעת השמה למשתנה (מספר כלשהו) והתכנית מדפיסה למסך את מספר האיברים במספר. ההנחה שמדובר במספרים חיוביים.
3. כתוב תכנית אשר מחשבת כמה שניות יש בשנה.
4. כתוב תכנית אשר מבצעת השמה למשתנה (מספר כלשהו) והתכנית מחשבת שורש ריבועי של המספר. במידה ואין שורש למספר אז תוצג הודעה מתאימה.