**Day1 of learning in Drop Hills**

**Basic syntax**

• PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web Pages.

• PHP is case sensitive, keywords as if, else, while, echo, classes, functions, and user-defined functions are not case-sensitive. But the variable is case sensitive.

• To get the data type of a variable, use the **var\_dump()** function.

• **var\_dump()** is a built-in PHP function that allows you to display information about the variable, including the type and value. It is commonly used for debugging purposes to inspect the contents of a variable.

• The **global** keyword is used to access a global variable from within a function.

• PHP also stores all global variables in an array called $GLOBALS [index]. The index holds the name of the variable. This array is also accessible from within functions and can be used to update global variables directly.

• The print\_r() function prints the information about a variable in a more human-readable way.

**Data Types**

• PHP Data Types o String

o Integer

o Float (Floating point numbers - also called double) o Boolean

o Array o Object o NULL

o Resource

• An Integer data type is a non-decimal number between -2,147,483,648 and 2,147,483,647.

• A float (floating point number) is a number with a decimal point or a number in exponential form.

• A Boolean represents two possible states: TRUE or FALSE. • An array stores multiple values in one single variable.

• Null is a special data type which can have only one value: NULL.

• A variable of data type NULL is a variable that has no value assigned to it.

• If a variable is created without a value, it is automatically assigned a value of NULL.

**Type Casting**

• Implicit Type Casting

o Implicit type conversion is commonly referred to as ‘Automatic Type Conversion’. It occurs automatically within the compiler without requiring external intervention from the user.

• Explicit Type Casting

o There are some cases where if the data types remain unchanged, it can give incorrect output. In such cases, typecasting can help to get the correct output and reduce the time of compilation. In explicit type casting, we have to force the conversion between data types. This type of casting is explicitly defined within the program.

**PHP Predefined Constants**

• PHP has nine predefined constants that change value depending on where they are used and therefore, they are called “magic constants”.

o \_\_CLASS\_\_: If used inside a class, the class name is returned. o \_\_DIR\_\_: The directory of the file.

o \_\_FILE\_\_: The file name including the full path.

o \_\_FUNCTION\_\_: If inside a function, the function name is returned. o \_\_LINE\_\_: The current line number.

o \_\_METHOD\_\_: If used inside a function that belongs to a class, both class and function name is returned.

o \_\_NAMESPACE\_\_: If used inside a namespace, the name of the namespace is returned.

o \_\_TRAIT\_\_: If used inside a trait, the trait name is returned.

**PHP Operators**

• Arithmetic Operators • Assignment Operators • Comparison Operators

• Increment/Decrement Operators • Logical Operators

• String Operators • Array Operators

• Conditional assignment Operators

**PHPArrays**

• An array is a special variable that can hold many values under a single name, and you can access the values by referring to and index number or name.

o Indexed Arrays

o Associative Arrays

o Multidimensional Arrays

**PHP Sorting Arrays**

• sort (): sort arrays in ascending order

• rsort (): sort arrays in descending order

• asort (): sort associative arrays in ascending order, according to the value • ksort (): sort associative arrays in ascending order, according to the key

• arsort (): sort associative arrays in descending order, according to the value • krsort (): sort associative arrays in descending order, according to the key

**PHP Global Variables – Superglobals**

• Some predefined variables in PHP are “superglobals”, which means that they are always accessible, regardless of scope – and you can access them from any function, class or file without having to do anything special.

• The PHP super global variables are: o $GLOBALS

o $\_SERVER o $\_REQUEST o $\_POST

o $\_GET o $\_FILES o $\_ENV

o $\_COOKIE o $\_SESSION

**OOP Concepts**

• OOP stands for Object-Oriented Programming.

• Object-oriented programming has several advantages over procedural programming: o OOP is faster and easier to execute

o OOP provides a clear structure for the programs

o OOP helps to keep the PHP code DRY “Don’t Repeat Yourself”, and makes the code easier to maintain, modify and debug.

o OOP makes it possible to create full reusable applications with less code and shorter development time.

**Class & Objects**

• Classes and objects are the two main aspects of object-oriented programming. • A class is a template for objects, and an object is an instance of a class.

• When the individual objects are created, they inherit all the properties and behaviors from the class, but each object will have different values for the properties.

**OOP– Inheritance**

• Inheritance in OOP = When a class derives from another class.

• The child class will inherit all the public and protected properties and methods from the parent class. In addition, it can have its own properties and methods.

• An inherited class is defined by using the extends keyword.

**OOP– Abstract Classes**

• Abstract classes and methods are when the parent class has named method, but need its child class(es) to fill out the tasks.

• An abstract class is a class that contains at least one abstract method. An abstract method is a method that is declared, but not implemented in the code.

• An abstract class or method is defined with the abstract keyword.

**OOP– Interfaces**

• Interfaces allow you to specify what methods a class should implement.

• Interfaces make it easy to use a variety of different classes in the same way. When one or more classes use the same interface, it is referred to as “polymorphism”.

• Interfaces are declared with the interface keyword.

**OOP– Traits**

• Traits are used to declare methods that can be used in multiple classes. Traits can have methods and abstract methods that can be used in multiple classes, and the methods can have any access modifier (public, private, or protected).

• Traits are declared with the trait keyword.

# Day2 of learning in Drop Hills

## GET vs POST

* Both GET and POST create an array (e.g., array(key1=> value1, key2 => value2, key3 => value3, …)). This array holds key/value pairs, where keys are the names of the form controls and values are the input data from the user.
* Both GET and POST are treated as $\_GET and $\_POST. These are superglobals, which means that they are always accessible, regardless of scope – and you can access them from any function, class or file without having to do anything special.
* $\_GET is an array of variables passed to the current script via the URL parameters.
* $\_POST is an array of variables passed to the current script via the HTTP POST method.

## When to use GET?

* Information sent from a form with the GET method is visible to everyone (all variable names and values are displayed in the URL). GET also has limits on the amount of information to send. The limitation is about 2000 characters. However, because the variables are displayed in the URL, it is possible to bookmark the page. This can be useful in some cases.
* GET may be used for sending non-sensitive data.
* **Note: GET should NEVER be used for sending passwords or other sensitive information.**

## When to use POST?

* Information sent from a form with the POST method is invisible to others (all names/values are embedded within the body of the HTTP request) and has no limits on the amount of information send.
* Moreover, POST supports advanced functionality such as support for multi-part binary input while uploading files to server.
* However, because the variables are not displayed in the URL, it is not possible to bookmark the page.

## PHP Date-Time

* The PHP date () function is used to format a date and/or a time.
* The PHP date () function formats a timestamp to a more readable date and time.
* Syntax: date (format, timestamp)

## Get a Date

* The required format parameter of the date () function specifies how to format the date( or time).
* Here are some characters that are commonly used for dates:
* d – represents the day of the month (01 to 31)
* m – represents a month (01 to 12)
* Y – represents a year (in four digits)
* l (lowercase ‘L’) – represents the day of the week

## Get a Time

* Here are some characters that are commonly used for times:
* H - 24-hour format of an hour (00 to 23)
* H – 12-hour format of an hour with leading zeros (01 to 12)
* i – Minutes with leading zeros (00 to 59)
* s – Seconds with leading zeros (00 to 59)
* a – Lowercase Ante meridiem and Post meridiem (am or pm)

## PHP include and require Statements

* The include (or require) statement takes all the text/code/markup that exists in the specified file and copies it into the file that uses the include statement.
* Including files is very useful when you want to include the same PHP, HTML, or text on multiple pages of a website.
* It is possible to insert the content of one PHP file into another PHP file (before the server executes it), with the include or require statement.
* The include and require statements are identical, except upon failure:
* Require will produce a fatal error (E\_COMPILE\_ERROR) and stop the script
* Include will only produce a warning (E\_WARNING) and the script will continue.

## PHP Cookies

* A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user’s computer. Each time the same computer requests a page with a browser, it will send the cookie too. With PHP, you can both create and retrieve cookie values.
* A cookie is created with the setcookie() function.
* Syntax: setcookie(name, value, expire, path, domain, secure, httponly); (only the name parameter is required. All other parameters are optional).

## PHP Sessions

* A session is a way to store information (in variables) to be used across multiple pages. Unlike a cookie, the information is not stored on the user’s computer.
* The internet there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn’t maintain state.
* Session variables solve this problem (addressed in point two) by storing user information to be used across multiple pages (e.g., username, favorite color, etc.). By default, session variables last until the user closes the browser.
* So, Session variables hold information about one single user, and are available to al pages in one application.
* A session is started with the session\_start() function.
* Session variables are set with the PHP global variable: $\_SESSION.

## PHP Filter

* PHP filters are used to validate and sanitize external input.
* The PHP filter extension has many of the functions needed for checking user input, and is designed to make data validation easier and quicker.

## PHP and JSON

* PHP has some built-in functions to handle JSON.
* json\_encode()
* json\_decode()

## What is JSON?

* JSON stands for JavaScript Object Notation, and is a syntax for storing and exchanging data.
* Since the JSON format is a text-based format, it can easily be sent to and from a server, and used as a data format by any programming language.

## PHP Exceptions

* An exception is an object that describes an error or unexpected behavior of a PHP script.
* Exceptions are thrown by many PHP functions and classes.
* User defined functions and classes can also throw exceptions.
* Exceptions are a good way to stop a function when it comes across data that it cannot use.

## Throwing an Exception

* The throw statement allows a user defined function or method to throw an exception. When an exception is thrown, the code following it will not be executed.
* If an exception is not caught, a fatal error will occur with an “Uncaught Exception” message.

## The try…catch…finally Statement

* The try…catch…finally statement can be used to catch exceptions. Code in the finally block will always run regardless of whether an exception was caught. If finally is present, the catch block is optional.

## The Exception Object

* The Exception Object contains information about the error or unexpected behavior that the function encountered.
* Syntax: new Exception (message, code, previous).

## PHP MySQL Database

* MySQL is a database system used on the web
* MySQL is a database system that runs on a server
* The data in a MySQL database are stored in tables. A table is a collection of related data, and it consists of columns and rows.
* A query is a question or a request.
* MySQLi extension (the “i” stands for improved)

# Day 3 Learning in DropHills

* Made a crud application for the Phone\_BOOK
* All the validation and verification were done.
* The validation and verification were all manually done

# Day 4 Learning in DropHills

## MD5 Hashing

* MD5 is a cryptographic hash function algorithm that takes the message as input of any length and changes it into a fixed-length message of 16 bytes. MD5 algorithm stands for the message-digest algorithm. MD5 was developed as an improvement of MD4, with advanced security purposes. The output of MD5(Digest size) is always 128 bits.

## Use of MD5 Algorithm

* It is used for file authentication.
* In a web application, it is used for security purposes. e.g., Secure password of users etc.
* Using this algorithm, we can store our password in 128 bits format.

## Application of MD5 Algorithm

* We use message digest to verify the integrity of files/authenticates files.
* MD5 was used for data security and encryption.
* It is used to Digest the message of any size and also used for Password verification.
* For Game Boards and Graphics.

## SHA1 Hashing

Sha-1 or Secure Hash Algorithm 1 is a cryptographic algorithm which takes an input and produces a 160-bit (20-byte) hash value. This hash value is known as a message digest. This message digest is usually then rendered as a hexadecimal number which is 40 digits long.

Both MD5 stands for Message Digest and SHA1 stands for Secure Hash Algorithm square measure the hashing algorithms wherever. The speed of MD5 is fast in comparison of SHA1’s speed.

However, SHA1 provides more security than MD5. The construct behind these hashing algorithms is that this square measure accustomed generate a novel digital fingerprint of knowledge or message that is understood as a hash or digest.

* The functions can’t be restrained.
* The size of the hash (or digest) is often fastened and doesn’t rely upon the scale of the info.
* No 2 distinct information set square measure able to manufacture the same hash.