1. **Difference b/w composer.json v/s composer.lock**

composer update:

composer update will update our dependencies as they are specified in composer.json.

For example, if our project uses this configuration:

```
"require": {
   "laravelcollective/html": "2.0.*"
}
```

Supposing we have actually installed the 2.0.1 version of the package, running composer update will cause an upgrade of this package (for example to 2.0.2, if it has already been released).

In detail composer update will:

- Read composer.json
- Remove installed packages that are no more required in composer.json
- Check the availability of the latest versions of our required packages
- Install the latest versions of our packages
- Update composer.lock to store the installed packages version

composer install :

composer install will install all of the dependencies as specified in the composer.lock file at the version specified (locked), without updating anything.

In detail:

- Read composer.lock file
- Install the packages specified in the composer.lock file

When to install and when to update#

- composer update is mostly used in the 'development' phase, to upgrade our project packages.
- composer install is primarily used in the 'deploying phase' to install our application on a production server or on a testing environment, using the same dependencies stored in the composer.lock file created by composer update.

Reference:
https://riptutorial.com/php/example/7331/difference-between--composer-install--and--composer-update-#:~:text=composer%20update%20is%20mostly%20used,dependencies%20stored%20in%20the%20composer.

2. **How to add module to composer**

ways to add the module.

1. Via composer

   composer require drupal/geofield

2. Download the zip file and upload in extend and install.

3. **How to run script using composer**
4. **How to create service.**
5. **Hoe to create rest API.**
6. **How to apply patch**
7. **How to display the content to the logged in user w/o using code.**
8. **How to translate the site and multilingual.**

   Step 1.

   Configuration > regional and language > Languages

   a. Add the required language.
   b. Save config.

   Step 2.

   Configuration > regional and language > Content Language

   a. Check the check box under custom language settings ex: taxonomy, media, content etc
   b. Save the config.

   Step 3.

   Configuration > regional and language > User interface translation

   a. Select the language from the list and apply the filter.
   b. Add the translation string.

9. **Login functionality**
10. **How to redirect the user to x page when they login.**
11. **How to check previous commit.**
12. **How to create the branch in git.**
13. **How to create the custom table.**
14. **Create custom block reference**

https://www.valuebound.com/resources/blog/drupal-8-how-to-create-a-custom-block-programatically

1. What are the versions you have used?

Ans:- D8

2. Drupal migration from Drupal 7 to Drupal 8

Ans;- > Install the migration module and enable it in D8 project, follow the instruction

3. How many projects have you done in Drupal 8?

Ans: - 3

4. Are your projects in multi-site set-up or standard-alone setup?

Ans:- Standalone

5. What are the domains you have worked till now?

Ans:- Laravel ,Cakephp , Drupal

6. Is there any interactions between users like login, …?

Ans :- using veeva id validation we are allowing users to access our websites .

7. Did you use only Drupal login or any third-party integration?

Ans :- yes , we are using third party api integration.

8. What is single sign-on?

Ans: Single sign-on (SSO) is an authentication method that enables users to securely authenticate with multiple applications and websites by using just one set of credentials.

9. Have you worked on a multi-language site?

Ans:- > yes , goto configuration -> Regional and language settings-> language translation-> add translation strings and if its dynamic content then add t() in twig files.

10. Where are you storing the converted multi-language strings/data?

Ans :- It will store in Database.(table name:-)

11. How do you approach upgrading your Drupal version?

Ans : by using $ composer update command

12. Are there any commands(composer/drush) to update versions?

Ans :- drush update , drush updatedb , composer update .

13. Have you created any custom modules in your project?

Ans:- Yes , Mandatory files :- .info file, .module file, library and route file is optional based on our requirements.

14. Have you used any hook functions?

Ans:- Yes. Hooks allow modules to alter and extend the behavior of Drupal core, or another module.

15. How can I define variables in hooks to send those to twig files?

Ans : hook_preprocess()

$variables['elements']['#id'] == 'elearningpopup'

16. In which particular hook I need to define variables?

Ans : hook_preprocess()

17. Have you worked on the views?

18. In my case I have one view and to expose the form in the ajax mode and can display it in the front-end side with two drop downs, when I change the country value, will that particular form be invoked in the form alter? Is that particular form invoked into the form_alter hook?

Ans:- using form id it will identify the dropdown.

19. What is the contextual filter in the filter?

Ans: Contextual filters in Drupal allow you to add or remove information (content) to the currently displayed view. The filter acts dynamically based on the URL. If the provided argument value is in the URL, the view will contain certain results, if the value is not in the URL, the view will be filtered with other results.

20. Have you used any third-party API integration?

Ans : zoom integration, Spotme, Brightcove

21. Do you have any idea how to get/post data from/to a third-party API?

22. How to display JSON data (like latest feeds/news) in our site which is coming from a third-party API?

23. Have you enabled any caching in your site? Have you used any contributed module for that?

24. Have you worked on any search APIs?

25. Did you use view search?

26. If you need to capture any information/form info from anonymous users? How can I capture

that information with some security UIs? What are the things we need to consider, whether we need to enable any modules?

27. What is the weight-in-dept in Drupal? Is that ordering/weightage available in the node level

or is it for taxonomy only?

Ans : - To set the priority to the form / blocks we use weights . -ve values will come in top and +ve values will settle in the bottom of the list.

28. Have you worked on any data visualization like data visualized options in your site like charts integration, maps integration?

29. How can you manage your tasks level? Have you used any Agile mechanism or Jira tickets?

30. Have you worked on paragraph content/module?

31. How can you define templates for the paragraph bundles?

32. When you start theming level, what are all things you should consider to start implementing new themes?

33. Is routing.yml dependent on theme level/module level?

Ans : module level

34. If you have written routing and controller files defined in the theme folder, in the future the user wants to make a new theme and you started implementing it but you forgot about that routing file. What will happen? Whether it will show an error or how it will be?

35. How can you process your project's deployments? Your site is already deployed to live and there are some enhancements you need to send to live and you have gone through QA also. How can you proceed with the deployment process?

36. If I have added new fields in a content-type, how can you deploy those to production/live?

37. Where can you export your configurations?

Home -> Administration -> Configuration -> Development -> config & synch

Ans : We can export our site configurations with drush command - $ drush cex

38. Assume that you have created a new content-type and you are exporting it to production. When you are exporting the new content-type, will it be automatically creating all the fields

or we are getting only the content-type structure?

Ans;- Only Structure will be exported.

39. Have you used drush/composer commands?

40. When we are using drush commands and when we are using composers?

Ans : Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. Composer is not a package manager in the same sense as Yum or Apt. Drush is a command line shell and Unix scripting interface for Drupal. Drush core ships with lots of useful commands for interacting with code like modules/themes/profiles. Similarly, it runs update.php, executes SQL queries and DB migrations, and misc utilities like run cron or clear cache.

41. If you want to include some library through composer command, how can you handle that one? For example, in the pdf generation, generally we use the dompdf or fpdfs, these we can include only from the composer.

42. Do you have composer.json file in your project? (From here we can define that particular name and we can run through the composer)

43. SESSION vs COOKIE

Ans : Cookies are only stored on the client-side machine, while sessions get stored on the client as well as a server.

44. Div vs span

Ans : A div element is used for block-level organization and styling of page elements, whereas a span element is used for inline organization and styling.

45. array_merge() vs array__combine()

Ans : array_merge() - This function merges the two or more arrays.

array_combine() - This array combines only two arrays.

46. unlink() vs unset()

Ans : unlink() is acting as the drop operation which will completely delete files, whereas, unset() is used to delete the file content for emptying it.

47. implode() vs explode()

Ans : The explode() function breaks a string into an array, but the implode() function returns a string from the elements of an array.

48. Memcache

Ans : Memcache improves Drupal application performance by moving standard caches out of the database and by caching the results of other expensive database operations. Note that Drupal doesn't support Memcache by default, and for this, we need to install it on the server.

49. Composer vs Drush

Ans : Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. Composer is not a package manager in the same sense as Yum or Apt. Drush is a command line shell and Unix scripting interface for Drupal.

50. Unique key vs Primary key

Ans : Primary key will not accept NULL values whereas Unique key can accept NULL values. A table can have only one primary key whereas there can be multiple unique keys on a table.

1. **What is the content type?**

A content type is a reusable collection of metadata (columns), workflow, behavior, and other settings for a category of items or documents in a Microsoft SharePoint Foundation 2010 list or document library. Content types enable you to manage the settings for a category of information in a centralized, reusable way.

Ans 2

A Drupal content type is a pre-defined structure for organizing and storing content in a website. Each content type can have its own fields, display settings, and custom behaviors. The two most common content types in Drupal are "Article" and "Basic page".

2. Can we create multiple content types?

Yes.

3. What is the block?

Blocks are ***boxes of content rendered into an area, or region***, of a web page.

4. Can we place multiple blocks in one region?

Drupal's block module is limited by the fact that a block can only have one instance. Each block has a 1:1 relationship with its region, weight, visibility (and other) settings. This means that it is impossible to have blocks in multiple regions or to have blocks that have different settings on different pages.

5. I want to place a block in the sidebar region that needs to be displayed in specific pages.

Explain it?

6. What are the Views?

Ans: The views module allows administrators and site designers to create, manage, and display lists of content. Each list managed by the view's module is known as a "view", and the output of a view is known as a "display".

7. What is Taxonomy?

Taxonomy, a powerful core module, allows you to connect, relate and classify your website's content. In Drupal, these terms are gathered within "vocabularies". The Taxonomy module allows you to create, manage and apply those vocabularies. Taxonomy is the practice of classifying content. It will come in handy for everything from menu and navigation schemes to view and display options.

8. How to show the list of Articles which are tagged to the particular Taxonomy? For example, I have 20 article pages and you have two taxonomies, taxonomy A and taxonomy B. We need to show the list of 8 articles which are tagged to the taxonomy B. Explain?

9. What is the Exposed filter?

10. If you expose a filter, will it show it to the users?

11. What are the hooks?

12. Which hook will help you to alter your form?

13. If we want to alter a form which is provided by a contributed module, should we directly write form_alter in that particular module?

14. Once you alter the form, I need to manage the data. How can we manage this altered data in the database?

15. If you alter the database table in your local, how will you send those changes to development?

16. What are the basic files to create a custom module?

17. What are the mandatory files to create a theme?

18. If we want to load js and css only on the specific page. How can you do that?

19. In the custom module, I want to alter the contributed module's functionality. What is the simple way to do that?

Ans:

https://drupal.stackexchange.com/questions/215010/how-do-i-include-javascript-on-a-single-page-in-a-way-that-is-amenable-to-scale

20. How will you fetch the data from the database into the custom module?

21. In the custom module, I have used one jQuery file. From that custom module I need to pass a variable to that js file. How can you do that?

22. If I want to expose the data to a third party from Drupal, either XML or JSON… How will you do that? (using web services we can export the data)

23. Have you used Composer and Drush?

24. What is the Composer?

25. What is the Drush?

26. I want to do some functionality while enabling the module or disabling the module. For example, my module needs a database. When I'm enabling the module, where do I need to write the code to create a table in the database?

27. What measurements are you using there in your project? Like Agile methodology…

28. Who will do the requirement analysis?

29. Who assigns you the work?

30. Did you have any situations where you have a lot in your work, like many tasks on your plate which you cannot handle?

31. Did you do the code review and all in your project?

32. Can you tell me one of the review comments that you got from your lead?

33. If you get a chance to work on different technology, would you accept it?

34. What have you done to learn new technologies? Did you earn any certifications or anything?

35. Did you work overtime?

36. Did you work on development projects and support as well?

37. What is your career goal? Let's say after 3 or 5 years?

38. Do you perform unit testing also?

39. When you do unit testing, where do you capture the results?

40. What is your sprint duration in your current project?

41. Are you part of any estimations and all in your project?

42. Do you know anything about estimations?

43. Can you tell me one example where you identified a critical bug/show blocker that you have identified?

44. Lets say, you are in the middle of the sprint, then the customer comes back, and says 'okay Akash, now the requirements/specifications are changed'. Would you accept it or would you reject it?

45. How many companies have you been working with?

46. Three companies in 3.6 years! That is too much, why is that?

47. If we onboard you, Would you serve long term or will you think of a better opportunity again?

48. Why did you select TCS, why not other companies? Are you giving interviews in other companies also?

49. Are you basically from Bangalore or from a different location?

50. Did you complete your studies and all?

51. How many days of the notice period?

52. Do you have any other opportunity in your hand?

53. Which location do you prefer?

54. What is your higher education?

55. Do you have any gaps in your education?

56. All these courses you have above 50%?

57. Do you have any gaps in your project?

1. what are all the things you worked in Drupal 8??

2. Have you worked with views??

Ans : Yes.

3. Why we use view in drupal 8?

Ans: The views module allows administrators and site designers to create, manage, and display lists of content. Each list managed by the views module is known as a "view", and the output of a view is known as a "display". Displays are provided in either block or page form, and a single view may have multiple displays. Optional navigation aids, including a system path and menu item, can be set for each page-based display of a view. By default, views may be created that list content (a Node view type), content revisions (a Node revisions view type) or users (a User view type). A view may be restricted to members of specific user roles, and may be added, edited or deleted at the views administration page

4. Have you used contextual filters and relationships in views??

5. Is it possible to use logged in user id in contextual filter??

Ans : Yes

6. What is form mode and view mode in drupal 8??

Ans: In Drupal, form mode and view mode, both are Display modes.Form modes allow for multiple sets of field widget orderings and customizations, just as view modes allow for different orderings and customization of field formatters.

7. How will you create the custom module ?? what is the structure??

Ans:

.info file

.module file

.install file

.routing.yml

8. Name the required parameters should mention in .info file??

Ans:

Notify Drupal about the existence of a module, theme, or install profile.

● Specify if this is a theme, or module.

● Provide information for the Drupal Web UI administration pages.

● Provide criteria to control module activation and deactivation and Drupal version compatibility.

● General administrative purposes in other contexts.

Ex: name: Hello World Module

description: Creates a page showing "Hello World".

package: Custom

type: module

core: 8.x

9. Have you created any drupal 9 sites??

10. Can you tell me about routing and controllers?? How will you create??

Ans:

Controller - The controller tells Drupal which method to call when someone goes to the URL for the page (which is defined in the route).

Route - A route is a path which is defined for Drupal to return some sort of content on. For

example, the default front page, '/node' is a route. When Drupal receives a request, it tries to match the requested path to a route it knows about.

11. Where will you create controller file ??

Ans: module/src/controller/controller.php

12. Have you used composer?? When you run the composer which files will be get updated??

Ans:- composer.json and composer.lock

13. Have you used layout builder module??

Ans: Layout Builder in Drupal 8 allows you to add/remove sections to display the content using different layouts, and customizing your pages based on the requirements.

Ref: https://www.specbee.com/blogs/layout-builder-drupal-8-complete-guide

14. What are the uses of content type in drupal 8??

15. Does translations works in drupal ?? explain??

Ans:- yes, add the t() in twig files for dynamic values and then add translation string in regional and language (admin->config->regional and lang translation -> language translation.)

16. What is hooks?? What are the types of hooks you worked on??

Ans: Hooks are how modules can interact with the core code of Drupal. They make it possible for a module to define new urls and pages within the site (hook_menu), to add content to pages (hook_block, hook_footer, etc.), to set up custom database tables (hook_schema) and more.

Generally, you can place hooks into one of three categories:

● Hooks that answer a question.

● Hooks that alter existing data.

● Hooks that react to an action.

17. Have you used preprocessor()?? Preprocess_hook()??

Ans: Preprocess functions allow Drupal themes to manipulate the variables that are used in Twig template files by using PHP functions to preprocess data before it is exposed to each template.

Syntax:

```
function hook_preprocess(&$variables, $hook) {
}
```

18. Have you used drupal behavior??

Ans: This ensures that our code will only run after the DOM has loaded and all elements are available. However with Drupal there is an alternate better method; using the functionality of Drupal.behaviors and once(). If used properly this will ensure that your code runs both on normal page loads and when data is loaded by AJAX (or BigPipe!) - but not jQuery methods like load() which should be avoided as Drupal behaviors will fail to load for loading functions other than ajax(). The Drupal.behaviors object is itself a property of the Drupal object, and when we want our module/theme to add new behaviors, the best method is to simply extend this object.

19. Types of array in php??

Ans: There are three types of arrays:

● Indexed arrays - Arrays with a numeric index.

● Associative arrays - Arrays with named keys.

- Multidimensional arrays - Arrays containing one or more arrays

20. Have you used push and pop in array?? When perform pop() which element is popped from the array??

Ans: The array_push method can be used to add or insert one or more items or elements from the end of an array. and The array_pop() method can be used to extract, delete and remove the elements/items from the end of the array.

21. How to get count of the elements of an array??

Ans: count($array);

22. How to find key is exists or not in an array??

Ans: array_key_exists(key, array);

23. How to convert string to array with space separator??

Ans: explode(“ ”, $str);

24. Have you used git?? Mention few commands??

Ans: Yes,

Git diff , git status, git add , git commit -m ”message”, git pull, git push, git checkout, git branch, git merge, git reflog, git reset, git reset --hard/–soft <commit-id>, git revert

25. What are the modules to be installed for multilingual site drupal 8?

Ans : The four core Drupal translation modules available :

- Language module.
- Content translation module.
- Interface translation module.
- Configuration translation module.

1. **What is taxonomy?**
2. **What are views?**
3. **Do you know about filters in views?**
4. **What are contextual filters?**
5. **What are Relationships?**
6. **Did you create any custom REST API?**
7. **What is content type?**

8. Composer command to export configurations
9. Tell me any 5 composer and drush commands you know
10. What is the difference between composer.json and composer.lock?
11. How do you create a custom module?
12. What are the required files to create a custom module?
13. What are the keys we need to add in the .info file?
14. What is the use of a controller?
15. Did you create any custom route?
16. How can we create a custom route?
17. How can you include/add libraries (js/css) in your module?
18. Do you know hooks?
19. What are hooks?
20. What is the use of hook_preprocessr?
21. What are twig filters?
22. What are plugins?
23. How to create a custom form?
24. Do you know Ajax?
25. Can you explain a scenario in which you have used ajax?
26. I have a form with few fields like News_title, description, country, state, ... Now I need to filter
27. news according to the selected country and state. And you need to show the latest news first.
28. How can you work this out with taxonomy?
29. Write logic to reverse an array without using built-in methods?
30. Write logic to print * pyramid in PHP?
31. What is normalization?
32. Explain the types of normalization?
33. What are JOINs?
34. Explain the types of JOINs?
35. Write query to get the second highest salary from the employee table
36. What are OOPs concepts?
37. What is inheritance?
38. How do you extend static class without creating object?
39. What is final class?
40. Ans: The final modifier for finalizing the implementations of classes, methods, and variables.
41. The main purpose of using a class being declared as final is to prevent the class from being
42. subclassed. If a class is marked as final then no class can inherit any feature from the final
43. class. You cannot extend a final class.

44. **What is the difference between static class and final class?**
45. **What are keys? What is the difference between primary key and unique key?**
46. **What is the interface?**
47. **Ans: An interface is a blueprint of a class. It has static constants and abstract methods. The**
48. **interface is a mechanism to achieve abstraction**
49. **How do you access/use the interface?**
50. **What is class?**
51. **Ans: A class is a user-defined type that describes what a certain type of object will look like. A**
52. **class description consists of a declaration and a definition. Usually these pieces are split into**
53. **separate files. An object is a single instance of a class. You can create many objects from the**
54. **same class type.**
55. **What is object?**
56. **What is abstraction?**
57. **Ans: Abstraction is the concept of object-oriented programming that "shows" only essential**
58. **attributes and "hides" unnecessary information. The main purpose of abstraction is hiding the**
59. **unnecessary details from the users.**
60. **Tell me a real world example for abstraction?**
61. **What are services in Drupal?**
62. **How to create service?**
63. **What is event subscriber?**
64. **What is event manager?**
65. **Do you know profiles in Drupal?**
66. **How to add comments section to profile like we have comments section in the blog or article**
67. **pages?**
68. **Did you work on migration?**
69. **What is multi-lingual site?**
70. **How many modules do we need to install for the multilingual site? What are those?**
71. **What do you know about cache?**
72. **Do you know how to setup Drupal project in local?**
73. **What is docker?**
74. **Explain the deployment process?**
75. **Which methodology you are following in your current organization? (Like Agile)**
76. **What is sprint?**

77. **If you need to complete the task within less than the given sprint, what should you do ? How**
78. **can you complete?**
79. **Who will do code reviews in your team?**
80. **How do you do unit testing?**
81. **What are the learning approaches you are following to learn Drupal?**

To get the Current user details

```
<?php

$current_user = \Drupal::currentUser();

//$account = \Drupal::currentUser(); //

//$user = \Drupal\user\Entity\User::load(\Drupal::currentUser()->id());

//print_r($user);die;

//var_dump($current_user);

$account = \Drupal\user\Entity\User::loadmultiple(); // pass your uid

$name = $account->getDisplayName();

//drupal_set_message($name);

//print_r($account);

$ids = \Drupal::entityQuery('user')// to get all users details

->condition('status', 1)

->condition('roles', 'administrator')

->execute();

$users = User::loadMultiple($ids);

foreach ($users as $user) {

$username = $user->get('name')->getString();

$mail = $user->get('mail')->getString();

$userlist[$mail] = $username;

}
```

1. http://drupalyourselves.blogspot.com/2017/03/tcs-interview-questions.html

2. https://drupalize.me/tutorial/overview-relationships-views?p=3377#:~:text=Relationships

%20are%20a%20powerful%20Views,data%20in%20our%20Drupal%20site

Drupal references

—-----------------------------

db_merge() :

https://coderwall.com/p/hfag3q/drupal-s-db_merge-function-does-either-an-update-or-an-insertdepending-on-whether-the-record-exists

1. **creation of custom theme and custom module**
2. **dependency injection**
3. **services**
4. **event subscriber**
5. **creation of custom form, plugins, custom block in drupal**
6. **different hook examples in drupal**
7. **different hook preprocess in drupal**
8. **caching in drupal**
9. **difference between drupal 7,8,9 and 10**
10. **performance in drupal**
11. **security modules in drupal**
12. **multilingual in drupal**
13. **different entities in drupal**
14. **drupal custom permissions**
15. **contextual filters, relationships in views**

```
<?php


/**

 * @file

 * Contains cambridge_product_group_menu.module.

 */


use Drupal\Core\Language\LanguageInterface;
```

```
/**
 * Hook_preprocess_page() to redirect users based on region.
 */
function cambridge_product_group_menu_preprocess_page(&$vars) {
  $vars['#attached']['library'][] =
'cambridge_product_group_menu/cambridge_product_group_menu';
}


/**
 * Implements hook_preprocess_menu().
 */
function cambridge_product_group_menu_preprocess_menu(&$variables, $hook) {
  // To get business unit name from the URL.
  $path = Drupal::request()->getPathInfo();
  $arg = explode('/', $path);
  $business_unit_name = $arg[2] == 'cambridge-english' ?
    'cambridge_english' : (array_key_exists('3', $arg) ? ($arg[3] == 'bibles' ?
$arg[3] : '') : ($arg[2] == "universitypress" ? 'academic' : $arg[2]));
  // Getting current langcode.
  $langcode = \Drupal::languageManager()-
>getCurrentLanguage(LanguageInterface::TYPE_CONTENT)->getId();
  // To get taxonomy bundle name from config form.
  $config = \Drupal::configFactory()-
>getEditable('cambridge_localization.countriescreationadminform');
  $bundle = $config->get($business_unit_name . '_bu');
  $query = \Drupal::entityTypeManager()->getStorage("taxonomy_term")->getQuery();
  $query->condition('vid', $bundle);
  $query->condition('field_country_code', $langcode, '=');
  $results = $query->execute();
  $tids = Drupal::entityTypeManager()->getStorage("taxonomy_term")-
>loadMultiple($results);
```

```
  if (count($tids) > 0) {
    foreach ($tids as $tdata) {
      $termId = $tdata->get('tid')->getValue();
      // To find the parent term Id.
      $parent = \Drupal::entityTypeManager()->getStorage('taxonomy_term')->loadParents($tdata->id());
      if (!is_null($parent)) {
        // To get the parent termId.
        foreach ($parent as $pid) {
          $parent_termId = implode(array_map('current', $pid->get('tid')->getValue()));
        }
      }
      if (!is_null($termId)) {
        $termId = implode(array_map('current', $termId));
      }
    }
  }
  $menu_name = in_array($business_unit_name, [
    'cambridge-english',
    'cambridge_english',
  ]) ? 'english' : ($business_unit_name == 'universitypress' ?
    'academic-menu' : $business_unit_name . '-menu');
  // To get the menu items.
  if ($variables['menu_name'] == $menu_name) {
    $new_menu = [];
    foreach ($variables['items'] as $item) {
      $domain_access_term_id = [];
      // To get the entity array elements value.
      $entity = \Drupal::entityTypeManager()->getStorage('menu_link_content')->loadByProperties(['uuid' => $item['original_link']->getDerivativeId()]);
      // To get the Domain access.
```

```
$below_new_menu = [];
      if (count($item['below']) > 0) {
        $below_new_menu = below_menu_items($item['below'], $below_new_menu,
$termId, $parent_termId);
        $below_sub_child_menu = [];
        foreach ($below_new_menu as $key => $sub_item) {
          if (count($sub_item['below']) > 0) {
            $below_sub_child_menu = below_menu_items($sub_item['below'],
$below_sub_child_menu, $termId, $parent_termId);
            // 3rd level menu
            $below_super_sub_child_menu = [];

            foreach ($below_sub_child_menu as $key => $super_sub_item) {
              if (count($super_sub_item['below']) > 0) {
                $below_super_sub_child_menu =
below_menu_items($super_sub_item['below'], $below_super_sub_child_menu, $termId,
$parent_termId);
              }
              $below_sub_child_menu[$key]['below'] = $below_super_sub_child_menu;
             // dd($below_super_sub_child_menu);

            }
            $below_new_menu[$key]['below'] = $below_sub_child_menu;
          }
         // dd($below_new_menu);
        }
      }
      $item['below'] = $below_new_menu;
      // dd($below_new_menu);
      foreach ($entity as $idMenu) {
        $menu_ids = $idMenu->get('field_domain_access')->getValue();
        if (!is_null($menu_ids)) {
          foreach ($menu_ids as $sub_tid) {
            $domain_access_term_id[] = $sub_tid['target_id'];
          }
        }
      }
      if (!is_null($domain_access_term_id)) {
        // To get the menu based on term id if country is enabled.
        if (in_array($termId, $domain_access_term_id)) {
          array_push($new_menu, $item);
        }
        // To get the menu based on region id or global is enabled.
        elseif (in_array($parent_termId, $domain_access_term_id)) {
          array_push($new_menu, $item);
        }
      }
    }
    if (!empty($new_menu)) {
      $variables['items'] = $new_menu;
    }
  }
}
function below_menu_items($items, $new_menu, $termId, $parent_termId) {
  foreach ($items as $key => $item) {
    $domain_access_term_id = [];
    // To get the entity array elements value.
    $entity = \Drupal::entityTypeManager()->getStorage('menu_link_content')-
>loadByProperties(['uuid' => $item['original_link']->getDerivativeId()]);
    // To get the Domain access.
    foreach ($entity as $idMenu) {
      $menu_ids = $idMenu->get('field_domain_access')->getValue();
      if (!is_null($menu_ids)) {
        foreach ($menu_ids as $sub_tid) {
          $domain_access_term_id[] = $sub_tid['target_id'];
        }
      }
    }
    if (!is_null($domain_access_term_id)) {
      // To get the menu based on term id if country is enabled.
      if (in_array($termId, $domain_access_term_id)) {
        //array_push($new_menu, $item);
        $new_menu[$key] = $item;
      }
      // To get the menu based on region id or global is enabled.
      elseif (in_array($parent_termId, $domain_access_term_id)) {
        // array_push($new_menu, $item);
        $new_menu[$key] = $item;
      }
    }
  }
  return $new_menu;
}
```

q1. What are the versions you have used? Ans:- D8

2. Drupal migration from Drupal 7 to Drupal 8

Ans;- > Install the migration module and enable it in D8 project, follow the instruction

3. How many projects have you done in Drupal 8?

Ans: - 3

4. Are your projects in multi-site set-up or standard-alone setup?

Ans:- Standalone

5. What are the domains you have worked till now?

Ans:- Laravel ,Cakephp , Drupal

6. Is there any interactions between users like login, …?

Ans :- using veeva id validation we are allowing users to access our websites .

7. Did you use only Drupal login or any third-party integration?

Ans :- yes , we are using third party api integration.

8. What is single sign-on?

Ans: Single sign-on (SSO) is an authentication method that enables users to securely authenticate with multiple applications and websites by using just one set of credentials.

9. Have you worked on a multi-language site?

Ans:- > yes , goto configuration -> Regional and language settings-> language translation-> add translation strings and if its dynamic content then add t() in twig files.

10. Where are you storing the converted multi-language strings/data?

Ans :- It will store in Database.(table name:-)

11. How do you approach upgrading your Drupal version?

Ans : by using $ composer update command

12. Are there any commands(composer/drush) to update versions?

Ans :- drush update , drush updatedb , composer update .

13.Have you created any custom modules in your project?

Ans:- Yes , Mandatory files :- .info file, .module file, library and route file is optional based on our requirements.

14.Have you used any hook functions?

Ans:- Yes. Hooks allow modules to alter and extend the behavior of Drupal core, or another module.

15.How can I define variables in hooks to send those to twig files?

Ans : hook_preprocess()

```
$variables['elements']['#id'] == 'elearningpopup'
```

16.In which particular hook I need to define variables?

Ans : hook_preprocess()

17.Have you worked on the views?

18.In my case I have one view and to expose the form in the ajax mode and can display it in the front-end side with two drop downs, when I change the country value, will that particular form be invoked in the form alter? Is that particular form invoked into the form_alter hook?

Ans:- using form id it will identify the dropdown.

19.What is the contextual filter in the filter?

Ans: Contextual filters in Drupal allow you to add or remove information (content) to the currently displayed view. The filter acts dynamically based on the URL. If the provided argument value is in the URL, the view will contain certain results, if the value is not in the URL, the view will be filtered with other results.

20. Have you used any third-party API integration?

Ans : zoom integration, Spotme, Brightcove

21. Do you have any idea how to get/post data from/to a third-party API?

22. How to display JSON data (like latest feeds/news) in our site which is coming from a third-party API?

23. Have you enabled any caching in your site? Have you used any contributed module for that?

24. Have you worked on any search APIs?

25. Did you use view search?

26. If you need to capture any information/form info from anonymous users? How can I capture that information with some security UIs? What are the things we need to consider, whether we need to enable any modules?

27. What is the weight-in-dept in Drupal? Is that ordering/weightage available in the node level or is it for taxonomy only?

Ans : - To set the priority to the form / blocks we use weights . -ve values will come in top and +ve values will settle in the bottom of the list.

28. Have you worked on any data visualization like data visualized options in your site like charts integration, maps integration?

29. How can you manage your tasks level? Have you used any Agile mechanism or Jira tickets?

30. Have you worked on paragraph content/module?

31. How can you define templates for the paragraph bundles?

32. When you start theming level, what are all things you should consider to start implementing new themes?

33. Is routing.yml dependent on theme level/module level?

Ans : module level

34. If you have written routing and controller files defined in the theme folder, in the future the user wants to make a new theme and you started implementing it but you forgot about that routing file. What will happen? Whether it will show an error or how it will be?

35. How can you process your project's deployments? Your site is already deployed to live and there are some enhancements you need to send to live and you have gone through QA also. How can you proceed with the deployment process?

36. If I have added new fields in a content-type, how can you deploy those to production/live?

37. Where can you export your configurations?

Home -> Administration -> Configuration -> Development -> config & synch

Ans : We can export our site configurations with drush command - $ drush cex

38. Assume that you have created a new content-type and you are exporting it to production. When you are exporting the new content-type, will it be automatically creating all the fields or we are getting only the content-type structure?

Ans;- Only Structure will be exported.

39. Have you used drush/composer commands?

40. When we are using drush commands and when we are using composers?

Ans : Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. Composer is not a package manager in the same sense as Yum or Apt. Drush is a command line shell and Unix scripting interface for Drupal. Drush core ships with lots of useful commands for interacting with code like modules/themes/profiles. Similarly, it runs update.php, executes SQL queries and DB migrations, and misc utilities like run cron or clear cache.

41. If you want to include some library through composer command, how can you handle that one? For example, in the pdf generation, generally we use the dompdf or fpdfs, these we can include only from the composer.

42. Do you have composer.json file in your project? (From here we can define that particular name and we can run through the composer)

43. SESSION vs COOKIE

Ans : Cookies are only stored on the client-side machine, while sessions get stored on the client as well as a server.

44. Div vs span

Ans : A div element is used for block-level organization and styling of page elements, whereas a span element is used for inline organization and styling.

45. array_merge() vs array_combine()

Ans : array_merge() - This function merges the two or more arrays. array_combine() - This array

combines only two arrays.

46. unlink() vs unset()

Ans : unlink() is acting as the drop operation which will completely delete files, whereas, unset() is used to delete the file content for emptying it.

47. implode() vs explode()

Ans : The explode() function breaks a string into an array, but the implode() function returns a string from the elements of an array.

48. Memcache

Ans : Memcache improves Drupal application performance by moving standard caches out of the database and by caching the results of other expensive database operations. Note that Drupal doesn't support Memcache by default, and for this, we need to install it on the server.

49. Composer vs Drush

Ans : Composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. Composer is not a package manager in the same sense as Yum or Apt. Drush is a command line shell and Unix scripting interface for Drupal.

50. Unique key vs Primary key

Ans : Primary key will not accept NULL values whereas Unique key can accept NULL values. A table can have only one primary key whereas there can be multiple unique keys on a table.

1. what are all the things you worked in Drupal 8??

2. Have you worked with views??

Ans : Yes

3. Why we use view in drupal 8?

Ans: The views module allows administrators and site designers to create, manage, and display lists of content. Each list managed by the views module is known as a "view", and the output of a view is known as a "display". Displays are provided in either block or page form, and a single view may have multiple displays. Optional navigation aids, including a system path and menu item, can be set for each page-based display of a view. By default, views may be created that list content (a *Node* view type), content revisions (a *Node revisions* view type) or users (a *User* view type). A view may be restricted to members of specific user roles, and may be added, edited or deleted at the views administration page

4. Have you used contextual filters and relationships in views??

5. Is it possible to use logged in user id in contextual filter??

Ans : Yes

6. What is form mode and view mode in drupal 8??

Ans: In Drupal, form mode and view mode, both are Display modes.Form modes allow for multiple sets of field widget orderings and customizations, just as view modes allow for different orderings and customization of field formatters.

7. How will you create the custom module ?? what is the structure??

Ans:

.info file

.module file

.install file

.routing.yml

8. Name the required parameters should mention in .info file ??

Ans:

Notify Drupal about the existence of a module, theme, or install profile.

1. Specify if this is a theme, or module.
2. Provide information for the Drupal Web UI administration pages.
3. Provide criteria to control module activation and deactivation and Drupal version compatibility.
4. General administrative purposes in other contexts.

Ex: name: Hello World Module

description: Creates a page showing "Hello World". package: Custom

type: module core: 8.x

9. Have you created any drupal 9 sites??

10. Can you tell me about routing and controllers?? How will you create??

Ans:

Controller - The controller tells Drupal which method to call when someone goes to the URL for the page (which is defined in the route).

Route - A route is a path which is defined for Drupal to return some sort of content on. For example, the default front page, '/node' is a route. When Drupal receives a request, it tries to match the requested path to a route it knows about.

11. Where will you create controller file ??

Ans: module/src/controller/controller.php

12. Have you used composer?? When you run the composer which files will be get updated??

Ans:- composer.json and composer.lock

13. Have you used layout builder module ??

Ans: Layout Builder in Drupal 8 allows you to add/remove sections to display the content using different layouts, and customizing your pages based on the requirements.

Ref: https://www.specbee.com/blogs/layout-builder-drupal-8-complete-guide

14. What are the uses of content type in drupal 8??

15. Does translations works in drupal ?? explain??

Ans:- yes, add the t() in twig files for dynamic values and then add translation string in regional and language (admin->config->regional and lang translation -> language translation.)

16. What is hooks?? What are the types of hooks you worked on??

Ans: Hooks are how modules can interact with the core code of Drupal. They make it possible for a module to define new urls and pages within the site (hook_menu), to add content to pages (hook_block, hook_footer, etc.), to set up custom database tables (hook_schema) and more.

Generally you can place hooks into one of three categories:

1. Hooks that answer a question.
2. Hooks that alter existing data.
3. Hooks that react to an action.

17. Have you used preprocessor()?? Preprocess_hook()??

Ans: Preprocess functions allow Drupal themes to manipulate the variables that are used in Twig template files by using PHP functions to preprocess data before it is exposed to each template.

Syntax:

```
function hook_preprocess(&$variables, $hook) {

}
```

18. Have you used drupal behavior??

Ans: This ensures that our code will only run after the DOM has loaded and all elements are available. However with Drupal there is an alternate better method; using the functionality of Drupal.behaviors and once(). If used properly this will ensure that your code runs both on normal page loads and when data is loaded by AJAX (or BigPipe!) - but not jQuery methods

like load() which should be avoided as Drupal behaviors will fail to load for loading functions other than ajax(). The Drupal.behaviors object is itself a property of the Drupal object, and when we want our module/theme to add new behaviors, the best method is to simply extend this object.

19. Types of array in php??

Ans: There are three types of arrays:

1. Indexed arrays - Arrays with a numeric index.
2. Associative arrays - Arrays with named keys.
3. Multidimensional arrays - Arrays containing one or more arrays

20. Have you used push and pop in array?? When perform pop() which element is popped from the array??

Ans: The array_push method can be used to add or insert one or more items or elements from the end of an array. and The array_pop() method can be used to extract, delete and remove the elements/items from the end of the array.

21. How to get count of the elements of an array??

Ans: count($array);

22. How to find key is exists or not in an array??

Ans: array_key_exists(key, array);

23. How to convert string to array with space separator??

Ans: explode(“ ”, $str);

24. Have you used git?? Mention few commands??

Ans: Yes,

Git diff , git status, git add , git commit -m ”message”, git pull, git push, git checkout, git branch, git merge, git reflog, git reset, git reset --hard/–soft <commit-id>, git revert

1. What is taxonomy?

2. What are views?

3. Do you know about filters in views?

4. What are contextual filters?

5. What are Relationships?

6. Did you create any custom REST API?

7. What is content type?

8. Composer command to export configurations

9. Tell me any 5 composer and drush commands you know

10. What is the difference between composer.json and composer.lock?

11. How do you create a custom module?

12. What are the required files to create a custom module?

13. What are the keys we need to add in the .info file?

14. What is the use of a controller?

15. Did you create any custom route?

16. How can we create a custom route?

17. How can you include/add libraries (js/css) in your module?

18. Do you know hooks?

19. What are hooks?

20. What is the use of hook_preprocessr?

21. What are twig filters?

22. What are plugins?

23. How to create a custom form?

24. Do you know Ajax?

25. Can you explain a scenario in which you have used ajax?

26. I have a form with few fields like News_title, description, country, state, ... Now I need to filter news according to the selected country and state. And you need to show the latest news first. How can you work this out with taxonomy?

27. Write logic to reverse an array without using built-in methods?

28. Write logic to print * pyramid in PHP?

29. What is normalization?

30. Explain the types of normalization?

31. What are JOINs?

32. Explain the types of JOINs?

33. Write query to get the second highest salary from the employee table

34. What are OOPs concepts?

35. What is inheritance?

36. How do you extend static class without creating object?

37. What is final class?

Ans: The final modifier for finalizing the implementations of classes, methods, and variables. The main purpose of using a class being declared as final is to prevent the class from being subclassed. If a class is marked as final then no class can inherit any feature from the final class. You cannot extend a final class.

38. What is the difference between static class and final class?

39. What are keys? What is the difference between primary key and unique key?

40. What is the interface?

Ans: An interface is a blueprint of a class. It has static constants and abstract methods. The interface is a mechanism to achieve abstraction

41. How do you access/use the interface?

42. What is class?

Ans: A class is a user-defined type that describes what a certain type of object will look like. A class description consists of a declaration and a definition. Usually these pieces are split into separate files. An object is a single instance of a class. You can create many objects from the same class type.

43. What is object?

44. What is abstraction?

Ans: Abstraction is the concept of object-oriented programming that “shows” only essential attributes and “hides” unnecessary information. The main purpose of abstraction is hiding the unnecessary details from the users.

45. Tell me a real world example for abstraction?

46. What are services in Drupal?

47. How to create service?

48. What is event subscriber?

49. What is event manager?

50. Do you know profiles in Drupal?

51. How to add comments section to profile like we have comments section in the blog or article pages?

52. Did you work on migration?

53. What is multi-lingual site?

54. How many modules do we need to install for the multilingual site? What are those?

55. What do you know about cache?

56. Do you know how to setup Drupal project in local?

57. What is docker?

58. Explain the deployment process?

59. Which methodology you are following in your current organization? (Like Agile)

60. What is sprint?

61. If you need to complete the task within less than the given sprint, what should you do ? How can you complete?

62. Who will do code reviews in your team?

63. How do you do unit testing?

64. What are the learning approaches you are following to learn Drupal?

PHP AND DRUPAL QUESTIONS

----

1) What is "echo" in PHP?

PHP echo output one or more string. It is a language construct not a function. So the use of parentheses is not required. But if you want to pass more than one parameter to echo, the use of parentheses is required.

Syntax: void echo (string $arg1 [, string $...])

2) What is "print" in PHP?

PHP print output a string. It is a language construct not a function. So the use of parentheses is not required with the argument list. Unlike echo, it always returns 1.

Syntax: int print (string $arg)

echo and print are more or less the same. They are both **used to output data to the screen**. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument.

Print_r function in PHP:-
Print_r function is used to display or print information, which is readable by humans.

3) What are magic constants in PHP?

Magic constants are the predefined constants in PHP which get changed on the basis of their use. They start with double underscore (__) and ends with double underscore.

1. __LINE__
2. __FILE__
3. __DIR__
4. __FUNCTION__
5. __CLASS__
6. __TRAIT__
7. __METHOD__
8. __NAMESPACE__
9. ClassName::clas

4) What is the array in PHP

An array is used to store multiple values in a single value

Indexed array: an array with a numeric key.

Associative array: an array where each key has its specific value.

Multidimensional array: an array containing one or more arrays within itself.

5) Explain some of the PHP array functions?

- array()
- array_chunk()
- count()
- sort()
- array_reverse()
- array_search()
- array_intersect()

6) What is the difference between indexed and associative array?

The indexed array holds elements in an indexed form which is represented by number starting from 0 and incremented by 1. For example:

```
$season=array("summer","winter","spring","autumn");
```

The associative array holds elements with name. For example:

```
$salary=array("Sonoo"=>"350000","John"=>"450000","Kartik"=>"200000");
```

7) Explain some of the PHP string functions?

There are many array functions in PHP:

- strtolower()
- strtoupper()
- ucfirst()
- lcfirst()
- ucwords()
- strrev()
- strlen()

8) What are the ways to include file in PHP?

- **include** only generates a warning, i.e., E_WARNING, and continue the execution of the script.
- **require** generates a fatal error, i.e., E_COMPILE_ERROR, and stop the execution of the script.

9) Explain setcookie() function in PHP?

PHP setcookie() function is used to set cookie with HTTP response. Once the cookie is set, you can access it by $_COOKIE superglobal variable

Reterive cookie syntax: `echo $_COOKIE ["user"];`

10) PHP Cookie

PHP cookie is a small piece of information which is stored at client browser. It is used to recognize the user.

Cookie is created at server side and saved to client browser. Each time when client sends request to the server, cookie is embedded with request. Such way, cookie can be received at the server side.

11) What is a session?

PHP session is used to store and pass information from one page to another temporarily (until user close the website).

PHP session technique is widely used in shopping websites where we need to store and pass cart information e.g. username, product code, product name, product price etc from one page to another.

session_start()

PHP $_SESSION is an associative array that contains all session variables. It is used to set and get session variable values. - $_SESSION

Example: Store information

```
$_SESSION["user"] = "Sachin";
```

Example: Get information

```
echo $_SESSION["user"];
```

```
session_destroy();
```

What is the method to register a variable into a session?

```
<?php
Session_register($ur_session_var);
?>
```

What is the difference between session and cookie?

The main difference between session and cookie is that *cookies are stored on user's computer in the text file* format while *sessions are stored on the server side*.

Cookies can't hold multiple variables, on the other hand, Session can hold multiple variables.

You can manually set an expiry for a cookie, while session only remains active as long as browser is open.

Is multiple inheritance supported in PHP?

PHP supports only single inheritance; it means that a class can be extended from only one single class using the keyword 'extended'.

What is *PDO* in PHP?

It is a set of PHP extensions that provide a core PDO class and database, specific drivers. It provides a vendor-neutral, lightweight, data-access abstraction layer. Thus, no matter what database we use, the function to issue queries and fetch data will be same. It focuses on data access abstraction rather than database abstraction.

What is the difference between `var_dump()` and `print_r()`

- The `var_dump` function displays structured information about variables/expressions including its **type** and **value**. Arrays are explored recursively with values indented to show structure. It also shows which array values and object properties are references.
- The `print_r()` displays information about a variable in a way that's readable by humans. array values will be presented in a format that shows keys and elements. Similar notation is used for objects.

Consider:

```
$obj          array 'qualitypoint'  'technologies'  'India'
```

`var_dump($obj)`will display below output in the screen:

And, `print_r($obj)` will display below output in the screen.

OOps concept

What is Object Oriented Programming?

Object oriented programming is a programming technique to design your application. Application can be of any type like it can be web based application, windows based application etc.

What is a class?

A class is a template for an object, a user-defined datatype that contains variables, properties, and methods.

Class represents all properties and behaviors of object.

What is an object?

Objects are created from Classes, is an instance of a class that is created dynamically.

Object in programming is similar to real word object. Every programming object has some properties and behaviors.

You can create object of class with the help of **new** keyword

What is Constructor and Destructor?

Constructor:

Constructor is a special type of function which will be called automatically whenever there is any object created from a class.

Destructor:

Destructor is a special type of function which will be called automatically whenever any object is deleted or goes out of scope.

What is Member Variable and Member function?

Member Variable – These are the variables defined inside a class. This data will be invisible to the outside of the class and can be accessed via member functions. These variables are called attribute of the object once an object is created.

Member function – These are the function defined inside a class and are used to access object data.

What is different types of Visibility? OR What are access modifiers?

Each method and property has its visibility. There are three types of visibility in PHP.

Types of visibility:

13. **public:** Public method or variable can be accessible from anywhere, Means a public method or variable of a class can be called outside of the class or in a subclass.
14. **protected:** A protected method or variable can only be called in that class & it's subclass.
15. **private:** A private method or variable of a class can only be called inside that class only in which it is declared.

What is Encapsulation?

Wrapping up member variables and methods together into a single unit (i.e. Class) is called Encapsulation.

16. Encapsulation is used to hide the values or state of a structured data object inside a class, preventing unauthorized parties' direct access to them.
17. Visibility is the mechanism for encapsulation.

```
class Person

{          private $name;

public function setName($name) {

$this->name = $name;

}

public function getName($name) {

return $this->name;

}

 }

$personObj = new Person();

$personObj->setName('Full Stack Tutorials');
```

```
$personObj->getName();
```

What is Abstraction?

Abstraction is a concept in which implementation details are hidden.

Abstraction is the process of hiding the internal details of an application from the outer world. Abstraction is used to describe things in simple terms. It’s used to create a boundary between the application and the client programs.

Abstraction is present in almost all the real life machines.

- Your car is a great example of abstraction. You can start a car by turning the key or pressing the start button. You don’t need to know how the engine is getting started, what all components your car has. The car internal implementation and complex logic is completely hidden from the user.
- We can heat our food in Microwave. We press some buttons to set the timer and type of food. Finally, we get a hot and delicious meal. The microwave internal details are hidden from us. We have been given access to the functionality in a very simple manner.

3. Abstraction in OOPS
Objects are the building blocks of Object-Oriented Programming. An object contains some properties and methods. We can hide them from the outer world through access modifiers. We can provide access only for required functions and properties to the other programs. This is the general procedure to implement abstraction in OOPS.

Abstract Class:

Abstract class are class which contains atleast one or more abstract method.

Abstract Method:

Abstract method is a method which is declared, but not defined.

18. PHP 5 introduces abstract classes and methods.
19. Classes defined as abstract may not be instantiated
20. Classes that contains at least one abstract method must also be abstract.

```
}
```

21. Methods defined as abstract simply declare the method's signature - they cannot define the implementation. Abstract methods cannot be defined as private.
22. Classes which are inheriting it's parent class must provides implementations for the abstract methods.

```
abstract class TV
{
private $isOn = false;
 abstract function getBrand();
 public function turnOnTV()
{
$this->isOn = true;
 }
 public function turnOffTV()
{
$this->isOn = false;
}
 }
 class Panasonic extends TV {
public function getBrand()
{
return "Panasonic";
}
 }
 class Sony extends TV {
 public function getBrand()
{
 return "Sony";
}
}
```

Q:- Can Class Properties be Abstract?

No. there is no need for a class property to be abstract. Class properties and constants are not implemented, instead, they are declared, which in a way already makes them abstract. They follow the same rules and are treated the same way when declared in an abstract class as they would any other class.

Note: All the remaining classes which are not abstract are called **Concrete Classes**

What is the need of abstract class?

Suppose we were modeling the behavior of animals, by creating a class hierarchy that started with a base class called Animal.

Animals are capable of doing different things like flying, digging and walking, but there are some common operations as well like eating and sleeping.

Some common operations are performed by all animals, but in a different way as well.

When an operation is performed in a different way, it is a good candidate for an abstract method (forcing subclasses to provide a custom implementation).

Explain about polymorphism?

It is simply "One thing, can use in different forms". Technically, it is the ability to redefine methods for derived classes.

#Example: One Class (Car) can extend two classes (Audi & BMW).

Types of Polymorphism?

Polymorphism could be static and dynamic both. Overloading is static polymorphism while, overriding is dynamic polymorphism.

23. Compile time polymorphism (Static) - Method Overloading
24. Runtime time polymorphism (Dynamic) - Method Overriding
25. **Overloading** is defining functions/methods that have same signatures with different parameters in the same class.

26. **Overriding** is redefining parent class functions/methods in child class with same signature. So, basically the purpose of overriding is to change the behavior of your parent class method.

What is the difference between Abstract class and Interface?

Abstract class	Interface
In abstract class a method must be declared as abstract. Abstract methods doesn’t have any implementation.	In interface all the methods by default are abstract.
Abstract class can also contain member variables and concrete functions/methods.	Interfaces cannot contain any member variables and concrete functions/methods except constants.
An Abstract methods can be declare with access modifiers like public, protected etc. Concrete Class which is extending the abstract class must be defined with the same or visibility.	All methods declared in an interface must be public.
A class can Inherits only one Abstract class and Multiple inheritance is not possible for Abstract class.	A class can implements many interfaces and Multiple interface inheritance is possible.
Only complete member of abstract class can be static.	Memebrs of Interface can not be static.

When to use abstract class and interface in PHP? Explain with real world Example?

Abstract class: Abstract Class is used when you something you know (Concrete Methods) and something which you don't know (Abstract Methods).

Interface: Interface is used when you don't know anything about implementation. (All methods are abstract)

What is final keyword?

PHP 5 introduces the final keyword, which prevents child classes from overriding a method by prefixing the definition with the final. If the class itself is being defined final then it cannot be extended.

What is STATIC keyword and what is it's use in PHP ?

27. Static Methods and Properties in PHP will be treated as public if no visibility is defined.
28. static can also be used to define static variables and for late static bindings.

What is Namespaces in PHP?

Namespaces is used to avoid conflicting definitions and enables more flexibility and organization in your code base.

What are the advantages of object oriented programming?

29. **Code Resusability:** it can be acheived through inheritance and traits
30. **Modularity:** it can be acheived through breaking large code into small modules, Modularity reduces complexity.
31. **Flexibility:** it can be acheived through polymorphism
32. **Maintainability:** it is to maintian code which follow Object Oriented Programming Concepts.
33. **Security:** it can be acheived through Encapsulation
34. **Testability:** it is easy to test.

List out some of the differences between a class and an object?

Answer

CLASS	OBJECT
It is a blueprint from which different instances (objects) are created.	Objects are the instances of the class.
Class is a logical entity.	Objects are physical entities.
Users can declare class only once.	Objects can be declared multiple times depending upon the requirements.
When a class is created there is no memory allocation	Objects allocated memory.

Class is a group of objects.	Objects are real-world entities such as pen, copy, mouse, etc.
Class can be declared using class keyword e.g class Student {}	Objects can be declared using the new keyword e.g. Student s1=new Student();

3 0

7. What are the main features of OOPs?

Answer

You have to list these four features of OOPs while answering this question-

- Inheritance
- Polymorphism
- Encapsulation
- Data Abstraction

8. What is Encapsulation in oops with an example?

Answer

The process of binding up data and functions together that manipulates them is known as encapsulation in OOPS. It protects the data from outside interference and misuse.

Let's understand the concept of encapsulation with both real-world examples and with the help of a program.

9. What is Polymorphism and its types in OOPS?

Answer

Polymorphism is one of the core concepts which is used in object-oriented programming. Polymorphism generally shows the relationships between parent class objects and child class objects.

Types of polymorphism in OOPs

- Compile Time Polymorphism- It is also known as Static Binding and allows the programmer to implement various methods. Names can be the same but their parameters should be different.
- Runtime Polymorphism- It is also known as Dynamic Binding and allows the programmer to call a single overridden method during the runtime of the program.

What is Inheritance?

Answer

It is a technique in which one class acquires the property of another class. With the help of inheritance, we can reuse the fields and methods of the existing class.

Inheritance has three type, are given below.

- Single inheritance
- Multiple inheritance
- Multi level inheritance

But PHP supports only single inheritance, where only one class can be derived from single parent class. We can also use multiple inheritance by using interfaces.

What is constructor and destructor?

Answer

Constructor and Destructor both are special functions which are automatically called when an object is created and destroyed.

19. What is traits? How it is used in php?

Answer

Traits is a group of methods that reuse in single inheritance. A Trait is intended to reduce some limitations of single inheritance by enabling a developer to reuse sets of methods.

11. $this is **a reserved keyword in PHP that refers to the calling object**.

Drupal

What is the purpose of Ctools in Drupal?

Ans: Ctools is a collection of APIs and tools aimed at making the developer experience better.

What exactly is PDO?

PHP Data Objects. It is a reliable way to access the database.

what are hooks?

. Explain how caching works in Drupal?

what is Drush

What is Drupal Cron?

What is a render array in Drupal?

Render arrays are the fundamental building blocks of Drupal content. Render arrays in Drupal give you a systematic approach to updating content programmatically before it is displayed.

What are the five conceptual layers in the Drupal system?

A. Data (nodes, etc.)
B. Modules
C. Blocks and menus
D. User permissions
E. Themes and templates

How the preprocessor works in Drupal?

It has the main role to set up the variable within the template (.tpl.php) files. In drupal version 7 onwards, you will get all templates and functions. But in Drupal 6, you won't get any templates and functions.

Explain Drupal cache?

Ans: Now once you are using any content management tools, this is going to store the content in local memory for the time being. Once the content is uploaded to the server, automatically, this will be deleted from the cache. It can store assembled pages and blocks for anonymous users.

In drupal, how to implement a contextual filter?

Ans: For every statistic, you need to create separate tables in the database. For say, you can store your webpage's nodes inside the node table. Even you can write comments and add extra users also with access.

Explain Drupal Taxonomy?

Ans: It helps to categorize the nodes on any particular website.

What do you mean by Drupal panels?

Ans: It's a module of drupal. The web administrator can customize layouts for different purposes. The content developers will be putting the contents inside these layers.

Refer ; https://www.adaface.com/blog/drupal-interview-questions/

https://www.drupal.org/node/2360331

PDO

solar search

panels

render array

View modes/display mode

Views

paragraphs

nodes

entity

block

contextual filter

exposed filter

configuration

Ctools

singleton design pattern drupal

==

add jquery in theme and css

.theme file

```
function matt_page_alter(&$page) {
  $page['#attached']['library'][] = 'core/jquery';
}
```

.librarirs.yml file

```
matt-corescripts:
  version: VERSION
  js:
    js/modernizr.custom.04204.js: {}
```

html.html.twig.

```
<link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css'>
```

We need to add external library (CSS, JS) into your_theme.libraries.yml and attach the library either to twig or in your controller

```
-----------------------------------------------------

myLibrary:

js:

https://stackpath.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js:    { type: external, minified: true }


css:

component:

https://stackpath.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css: { type: external, minified: true }

{{ attach_library ('mythemename/mylibrary') }}
```

```
debug

=======

$node = node_load(123);

print_r($node);

?>
```

Drupal devel module provides dsm and dpm functions to debug code within drupal.

```
$node = node_load(123);

dpm($node);

?>
```

contextual filter

==================

When do I need a contextual filter vs. a relationship?

Ans:

A relationship is a join.

Acceptd Just think of it terms of SQL:

a ‘relationship’ is a JOIN on another table

a ‘filter’ (contextual or not) is a WHERE clause.

‘Contextual’ just means the parameter comes from the current context, e.g. part of the URL, something in the session,

I think you described it well yourself. A relationship is a join. You are joining another table to the base table for the view you are creating. A contextual filter is an argument you are passing to the query created by the view.

If you are creating a list of node titles a simplified query you are creating might be:

```
SELECT title FROM {node};
```

A contextual filter would pass an argument into that query. You can grab the argument from the URL or a number of other places. In this example the type would be an argument you are passing using a contextual filter:

SELECT title FROM {node} WHERE type = ‘page’;

An example of a relationship might be accessing user info about the node author. If you added a relationship to the user on the user id the query might look something like this:

SELECT title n, username u FROM {node} n JOIN {user} u ON n.uid = u.uid WHERE type = ‘page’;

==
==

1) **Entity is a building blocks having unified structure and it is fieldable**

   Example of entites are nodes, user, taxonomy

2)

Disable cache for a block in Drupal 8

For core $vars[‘#cache’][‘max-age’]=0

For custom getCacheMaxAge()

`\Drupal::service('page_cache_kill_switch')->trigger();` to disable the cache for anonymous users and all others too.