1.Explain what Laravel's query builder is and how it provides a simple and elegant way to interact with databases.

The Laravel framework's database query builder offers a convenient and fluent interface for creating and operating database queries. It can be used to accomplish most database operations in the application and performs on all Laravel Framework-supported database systems. It is based on the PDO parameter binding to protect the application against any SQL injection attacks. The advantage on gets here is that there is no need to clean the strings being passed as bindings.

For example:

To fetch data from database:

$users = DB::table('users')->get();

Or to fetch data according to the user Id

$user_id = DB::table('users')->where('name', 'John')->pluck('id');

2.Write the code to retrieve the "excerpt" and "description" columns from the "posts" table using Laravel's query builder. Store the result in the $posts variable. Print the $posts variable.

$posts= DB::table('posts')->select('excerpt', 'description')->get();

3.Describe the purpose of the distinct() method in Laravel's query builder. How is it used in conjunction with the select() method?

Distinct() method is used because of there is any duplicate data in the database, this method pick one data from the table. Without using this method, it will show all data including duplicate data.

For example

$query = DB::table('products')->distinct()->get(['products_name']);

4.Write the code to retrieve the first record from the "posts" table where the "id" is 2 using Laravel's query builder. Store the result in the $posts variable. Print the "description" column of the $posts variable.

$posts=DB::table('posts')->where('id', 2)->first();

$posts = DB::table('posts')->select('description')->get();

5.Write the code to retrieve the "description" column from the "posts" table where the "id" is 2 using Laravel's query builder. Store the result in the $posts variable. Print the $posts variable.

$posts = DB::table('posts')->select('description')->where('id', 2);

6.Explain the difference between the first() and find() methods in Laravel's query builder. How are they used to retrieve single records?

First() method is use for fetch first record only from the particular table of the particular column

For example:

$posts=DB::table('posts')->where('id', 2)->first();

Find() method is use to find particular data directly from its id, no need to put where()

For example:

$posts=DB::table('posts')->find(2);

7.Write the code to retrieve the "title" column from the "posts" table using Laravel's query builder. Store the result in the $posts variable. Print the $posts variable.

$posts=DB::table('posts')->select('title')->get( );

8.Write the code to insert a new record into the "posts" table using Laravel's query builder. Set the "title" and "slug" columns to 'X', and the "excerpt" and "description" columns to 'excerpt' and 'description', respectively. Set the "is_published" column to true and the "min_to_read" column to 2. Print the result of the insert operation.

```
$result = DB::table('posts')->insert([

    'title' => 'X',

    'slug' => 'X',

    'excerpt' => 'excerpt',

    'description' => 'description',

    'is_published' => true,

    'min_to_read' => 2,

]);


if ($result) {

    echo "Record inserted successfully.";

} else {

    echo "Failed to insert record.";

}
```

9.Write the code to update the "excerpt" and "description" columns of the record with the "id" of 2 in the "posts" table using Laravel's query builder. Set the new values to 'Laravel 10'. Print the number of affected rows.

10.Write the code to delete the record with the "id" of 3 from the "posts" table using Laravel's query builder. Print the number of affected rows.

$delete = DB::table('posts')->where('id', 3)->delete();

11.Explain the purpose and usage of the aggregate methods count(), sum(), avg(), max(), and min() in Laravel's query builder. Provide an example of each.

Count() method in Laravel query builder is used to retrieve the number of records from a specific query. For example, $count= DB::table('posts')->where('is_published', true)->count();

Sum() method used in Laravel query builder to calculate specific number columns value. For example, $sum= DB::table('posts')->where('is_published', true)->sum('id');

Avg() method used in Laravel query builder to find the average of specific number columns value For example, $avg= DB::table('posts')->where('is_published', true)->avg('id');

Max() this method used for retrive maximum value from a specific column for example,

$maxPrice = DB::table('products')->max('price');

Min() this method used for retrive minimum value from a specific column for example,

$maxPrice = DB::table('products')->min('price');

12.Describe how the whereNot() method is used in Laravel's query builder. Provide an example of its usage.

whereNot() method used there is to add not equal condition in a complex query. For example, $users = DB::table('users')->whereNot('status', 'active')->get();

13.Explain the difference between the exists() and doesntExist() methods in Laravel's query builder. How are they used to check the existence of records?

Exitst() method used to check where data is exist or nit

14.Write the code to retrieve records from the "posts" table where the "min_to_read" column is between 1 and 5 using Laravel's query builder. Store the result in the $posts variable. Print the $posts variable.

$results = DB::table('posts')->whereBetween('min_to_read', [1, 5])->get();

15.Write the code to increment the "min_to_read" column value of the record with the "id" of 3 in the "posts" table by 1 using Laravel's query builder. Print the number of affected rows.

$rows = DB::table('posts')->where('id', 3)->increment('min_to_read', 1);