

## Main menu

[Skip to content](#)

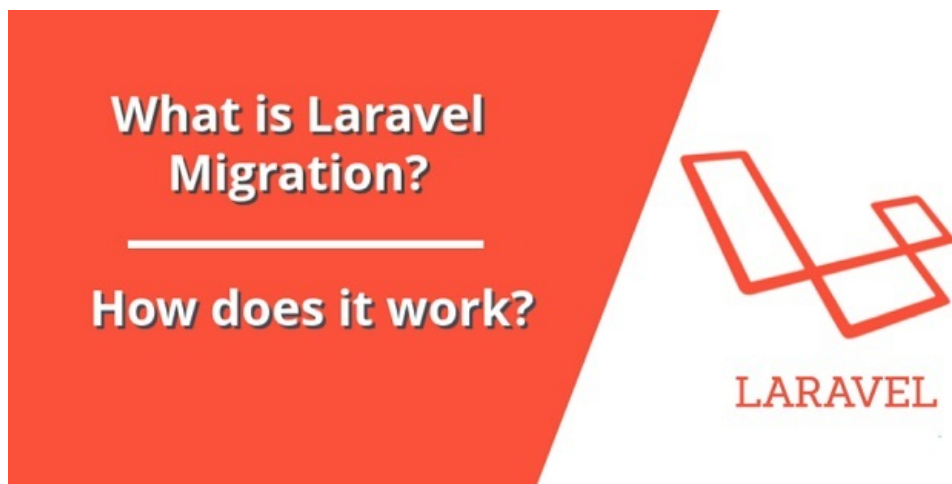
- [Home](#)
- [ক্যারিয়ার গাইডলাইন](#)
  - [Web Design এবং Web Development কি?](#)
  - [ZCPE কি?](#)
  - [Laravel Framework কি?](#)
  - [Industrial attachment এবং Industrial Training কি?](#)
  - [Computer Networking এবং Server Administration কি?](#)
  - [মোবাইল অ্যাপ ডেভেলপমেন্ট এক উজ্জ্বল ক্যারিয়ারের হাতছানি!](#)
- [ট্রেনিংস](#)
  - [Web Development](#)
    - [Responsive Web Design With WordPress Theme Customization](#)
    - [Professional Web Development with PHP, JAVASCRIPT, MySQL and WordPress Theme Development](#)
    - [Zend Certified PHP Engineering \(ZCPE\) Course](#)
    - [Professional WordPress Plugin Development Beginning to Advance](#)
    - [Laravel Beginning to Advance Course](#)
    - [JavaScript Beginning to Advance Course](#)
  - [Networking And Server Administration](#)
    - [Cisco](#)
      - [CCNA Certification](#)
    - [ISP Setup and Administration using MikroTik](#)
    - [Windows 2016 Server Administration](#)
    - [REDHAT ISP Setup](#)
  - [Mobile Apps Development](#)
    - [Native Android Apps Development with Java](#)
- [About Us](#)
- [Contact Us](#)

Menu

Posted on [December 29, 2017](#) by [Masud Alam](#)

## বাংলায় PHP LARAVEL FRAMEWORK পর্ব-৮: LARAVEL Migrations

### Laravel Framework এ Migration কি?



#### What is Laravel Migration

প্রযুক্তিগত অগ্রগতির সাথে সাথে , আমরা যারা আই.টি. পেশাদার, ওয়েব ডেভেলপার অথবা প্রোগ্রামার, আমরা অনেকেই 'Database Schema' নামটির সাথে পরিচিত এবং আমরা অনেকেই Database Schema কে create, modify এবং up-to-date রাখতে গিয়ে অনেক সময়ের সম্মুখীন হই। আপনার Database কে up-to-date রাখা এবং সব জটিলতা সমাধান করার জন্য, Laravel Framework নিয়ে আসছে Migration সুবিধা। Laravel Framework এ Migration ফাইল গুলি /database/migrations ফোল্ডারে থাকে।

Laravel Framework এ, Migrations হচ্ছে আপনার Database Schema গুলোর যেকোনো গঠন পরিবর্তন করার জন্য একটি ফাইল-ভিত্তিক পদ্ধতি। আপনার Database এর Version Control System এর মতো। যা দিয়ে আপনার টিমকে অ্যাপ্লিকেশন গুলোর Database Schema কে সহজেই create, modify এবং share করার সুযোগ দিতে পারেন। এবং current schema কে up-to-date রাখতে পারেন। যেমন, আপনি আপনার Database এ table তৈরি এবং Drop করা, table এ নতুন column যোগ করা, আপডেট করা, মুছে ফেলা, table এ index যোগ করা সহ অন্যান্য কাজের জন্য Laravel Migration ব্যবহার করতে পারেন। পরবর্তীতে যেকোনো ভুলের জন্য অথবা যেকোনো পরিবর্তনের জন্য আপনি খুব সহজে আগের অবস্থায় অথবা পিছনের যেকোনো ভার্শন এ ফিরে আসতে পারেন। কারণ প্রতিটি মাইগ্রেশন একটি টেক্সট ফাইল সংরক্ষণ করা হয়, আপনি আপনার Project Repository তে রেখে তাদের পরিচালনা করতে পারেন। Laravel Framework এ Migrations সাধারণত Laravel এর schema builder সঙ্গে সংযুক্ত থাকে, যাতে সহজেই আপনার অ্যাপ্লিকেশন এর Database Schema তৈরি করতে পারেন।

ধরুন, আপনি এবং আপনার টিম git Version Control ব্যবহার করেন। হঠাৎ আপনার মনে হলো user টেবিল এ নতুন একটি ফিল্ড লাগবে যার নাম status এবং টাইপ হবে boolean। এখন আপনি Laravel Framework এর migration সুবিধা ব্যবহার করে টেবিলের পরিবর্তন টা করলেন ও git এ পাঠিয়ে দিলেন। এখন আপনার team এর অন্যান্য সদস্যরা চাইলে git merge করে Laravel migration ফাইলটি রান করে Database কে up-to-date করে নিতে পারেন। আগের মতো আর .sql ফাইল গুলো একজন আরেকজনের কাছ থেকে খুঁজে নিতে হবেনা।

## Migration নিয়ে কাজ করার পূর্ব প্রস্তুতি :

Laravel Framework এ Migration নিয়ে কাজ করতে হলে আপনাকে প্রথমে Database এর host এর সাথে connect এবং Database Select করতে হবে। আর এর জন্য আপনাকে আপনার project এর root ফোল্ডারের .env ফাইল এ Database এর host, username, password এবং Database name বলে দিতে হবে। নিচের ছবিটি লক্ষ্য করুন :

```
7 DB_CONNECTION=mysql
8 DB_HOST=127.0.0.1
9 DB_PORT=3306
10 DB_DATABASE=laravel
11 DB_USERNAME=root
12 DB_PASSWORD=''
```

Laravel Database Connection in .env file-

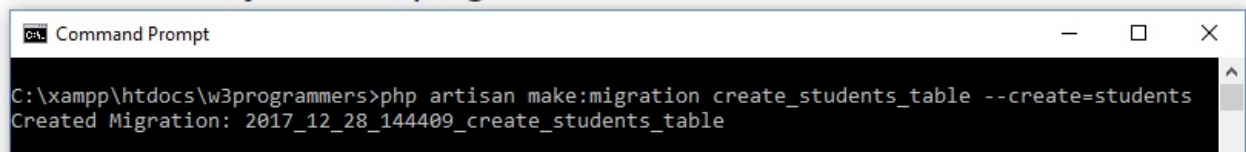
**Note:** .env file edit করার পর অবশ্যই php artisan সার্ভার টি রিস্টার্ট দিতে হবে। অথবা "php artisan config:clear" এই command টি run করতে হবে।

## কিভাবে Laravel Framework এ নতুন Migration File তৈরি করা হয় ?

Laravel Framework এ migration file তৈরি করতে হলে আপনাকে Terminal এ আপনার Project এর root এর ভেতর ঢুকে migration command রান করতে হবে। ধরুন, আমরা students table তৈরি করা জন্য একটা migration file বানাব। তাহলে আমাদের CLI তে artisan command টি হবে ঠিক নিচের মত:

```
1 | php artisan make:migration create_students_table --create=students
```

এখন CLI নিচের মতো ফলাফল দেখাবে :



```
C:\xampp\htdocs\w3programmers>php artisan make:migration create_students_table --create=students
Created Migration: 2017_12_28_144409_create_students_table
```

এখন আপনি database/migrations ফোল্ডারে **2017\_12\_28\_144409\_create\_students\_table.php** নামে নতুন তৈরী হওয়া একটা file দেখতে পারবেন। এবং ফাইলটি খুললে ঠিক নিচের মতো কোডগুলো পাবেন:

```
1 <?php
2
3 use Illuminate\Support\Facades\Schema;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Database\Migrations\Migration;
6
7 class CreateStudentsTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('students', function (Blueprint $table) {
17             $table->increments('id');
18             $table->timestamps();
```

```

19     });
20 }
21
22 /**
23  * Reverse the migrations.
24  *
25  * @return void
26  */
27 public function down()
28 {
29     Schema::dropIfExists('students');
30 }
31 }

```

**ব্যাখ্যা :** প্রথমত এখানে schema, Blueprint ও Migration নামের তিনটি ক্লাসকে ব্যবহার করা হয়েছে। এবং Migration নামের ক্লাস কে extend করে CreateStudentsTable নামের একটি ক্লাস তৈরী করা হয়েছে। CreateStudentsTable class এ আমরা দুটি Method দেখতে পাই, একটি up অন্যটি down । up মেথড দিয়ে আমরা নতুন table তৈরী, table এ data insert, table এ নতুন field যোগ এবং অন্য যেকোনো পরিবর্তনও করতে পারি। আর down মেথড এ আমরা up মেথডে যে পরিবর্তন টা করেছি সেটা কিভাবে undo করা যায় সেই প্রসেসটা বলে দিতে পারি এবং table drop ও করে দিতে পারি। সাধারণত যেকোনো Table এ একটি id থাকা লাগে এবং table এর যেকোনো রেকর্ড করে তৈরি ও পরিবর্তন হয়েছে সেইটাও জানতে হয়, তাই Laravel Framework আগেই আমাদের জন্য এই বেসিক বিষয় গুলো লিখে দিয়েছে। কিন্তু আমাদের প্রয়োজন না হলে এগুলো মুছে ফেলতে পারি।

## Migration ব্যবহার করে Database এ নতুন Table তৈরি করা :

আমরা আমাদের students table টায় আরও কিছু ফিল্ড/কলাম(Column) যোগ করব। আর তার জন্য Laravel blueprint class এর \$table object দিয়ে database table এর জন্য বিভিন্ন data type সহ \$table object এর বিভিন্ন Method call করব। লাইন নম্বর ১৬ থেকে ২১ দেখুন :

```

1  <?php
2  use Illuminate\Support\Facades\Schema;
3  use Illuminate\Database\Schema\Blueprint;
4  use Illuminate\Database\Migrations\Migration;
5
6  class CreateStudentsTable extends Migration
7  {
8      /**
9       * Run the migrations.
10      *
11      * @return void
12      */
13      public function up()
14      {
15          Schema::create('students', function (Blueprint $table) {
16              $table->increments('id');
17              $table->string('name', 20);
18              $table->string('email', 50);
19              $table->string('mobile', 20);
20              $table->tinyInteger('status')->default(1);
21              $table->timestamps();
22          });
23      }
24
25      /**
26       * Reverse the migrations.
27       *
28       * @return void
29       */
30      public function down()
31      {
32          Schema::dropIfExists('students');
33      }
34  }

```

**Note:** Database table এ কলামগুলো তৈরি করার জন্য যে মেথড গুলো ব্যবহার করা হয়েছে সেগুলোকে বলে কলাম মেথড(Column methods)। নিজে আপনাদের সুবিধার্থে সবগুলো কলাম মেথড এর [laravel.com](https://laravel.com) থেকে copy করে দেওয়া হলো :

Command	Description
\$table->bigIncrements('id');	Auto-incrementing UNSIGNED BIGINT (primary key) equivalent column.
\$table->bigInteger('votes');	BIGINT equivalent column.
\$table->binary('data');	BLOB equivalent column.
\$table->boolean('confirmed');	BOOLEAN equivalent column.
\$table->char('name', 100);	CHAR equivalent column with an optional length.
\$table->date('created_at');	DATE equivalent column.
\$table->dateTime('created_at');	DATETIME equivalent column.
\$table->dateTimeTz('created_at');	DATETIME (with timezone) equivalent column.
\$table->decimal('amount', 8, 2);	DECIMAL equivalent column with a precision (total digits) and scale (decimal digits).

<code>\$table-&gt;double('amount', 8, 2);</code>	DOUBLE equivalent column with a precision (total digits) and scale (decimal digits).
<code>\$table-&gt;enum('level', ['easy', 'hard']);</code>	ENUM equivalent column.
<code>\$table-&gt;float('amount', 8, 2);</code>	FLOAT equivalent column with a precision (total digits) and scale (decimal digits).
<code>\$table-&gt;geometry('positions');</code>	GEOMETRY equivalent column.
<code>\$table-&gt;geometryCollection('positions');</code>	GEOMETRYCOLLECTION equivalent column.
<code>\$table-&gt;increments('id');</code>	Auto-incrementing UNSIGNED INTEGER (primary key) equivalent column.
<code>\$table-&gt;integer('votes');</code>	INTEGER equivalent column.
<code>\$table-&gt;ipAddress('visitor');</code>	IP address equivalent column.
<code>\$table-&gt;json('options');</code>	JSON equivalent column.
<code>\$table-&gt;jsonb('options');</code>	JSONB equivalent column.
<code>\$table-&gt;lineString('positions');</code>	LINESTRING equivalent column.
<code>\$table-&gt;longText('description');</code>	LONGTEXT equivalent column.
<code>\$table-&gt;macAddress('device');</code>	MAC address equivalent column.
<code>\$table-&gt;mediumIncrements('id');</code>	Auto-incrementing UNSIGNED MEDIUMINT (primary key) equivalent column.
<code>\$table-&gt;mediumInteger('votes');</code>	MEDIUMINT equivalent column.
<code>\$table-&gt;mediumText('description');</code>	MEDIUMTEXT equivalent column.
<code>\$table-&gt;morphs('taggable');</code>	Adds taggable_id UNSIGNED INTEGER and taggable_type VARCHAR equivalent columns.
<code>\$table-&gt;multiLineString('positions');</code>	MULTILINESTRING equivalent column.
<code>\$table-&gt;multiPoint('positions');</code>	MULTIPOINT equivalent column.
<code>\$table-&gt;multiPolygon('positions');</code>	MULTIPOLYGON equivalent column.
<code>\$table-&gt;nullableMorphs('taggable');</code>	Adds nullable versions of morphs() columns.
<code>\$table-&gt;nullableTimestamps();</code>	Alias of timestamps() method.
<code>\$table-&gt;point('position');</code>	POINT equivalent column.
<code>\$table-&gt;polygon('positions');</code>	POLYGON equivalent column.
<code>\$table-&gt;rememberToken();</code>	Adds a nullable remember_token VARCHAR(100) equivalent column.
<code>\$table-&gt;smallIncrements('id');</code>	Auto-incrementing UNSIGNED SMALLINT (primary key) equivalent column.
<code>\$table-&gt;smallInteger('votes');</code>	SMALLINT equivalent column.
<code>\$table-&gt;softDeletes();</code>	Adds a nullable deleted_at TIMESTAMP equivalent column for soft deletes.
<code>\$table-&gt;softDeletesTz();</code>	Adds a nullable deleted_at TIMESTAMP (with timezone) equivalent column for soft deletes.
<code>\$table-&gt;string('name', 100);</code>	VARCHAR equivalent column with a optional length.
<code>\$table-&gt;text('description');</code>	TEXT equivalent column.
<code>\$table-&gt;time('sunrise');</code>	TIME equivalent column.
<code>\$table-&gt;timeTz('sunrise');</code>	TIME (with timezone) equivalent column.
<code>\$table-&gt;timestamp('added_on');</code>	TIMESTAMP equivalent column.
<code>\$table-&gt;timestampTz('added_on');</code>	TIMESTAMP (with timezone) equivalent column.
<code>\$table-&gt;timestamps();</code>	Adds nullable created_at and updated_atTIMESTAMP equivalent columns.
<code>\$table-&gt;timestampsTz();</code>	Adds nullable created_at and updated_atTIMESTAMP (with timezone) equivalent columns.
<code>\$table-&gt;tinyIncrements('id');</code>	Auto-incrementing UNSIGNED TINYINT (primary key) equivalent column.
<code>\$table-&gt;tinyInteger('votes');</code>	TINYINT equivalent column.
<code>\$table-&gt;unsignedBigInteger('votes');</code>	UNSIGNED BIGINT equivalent column.
<code>\$table-&gt;unsignedDecimal('amount', 8, 2);</code>	UNSIGNED DECIMAL equivalent column with a precision (total digits) and scale (decimal digits).
<code>\$table-&gt;unsignedInteger('votes');</code>	UNSIGNED INTEGER equivalent column.
<code>\$table-&gt;unsignedMediumInteger('votes');</code>	UNSIGNED MEDIUMINT equivalent column.
<code>\$table-&gt;unsignedSmallInteger('votes');</code>	UNSIGNED SMALLINT equivalent column.
<code>\$table-&gt;unsignedTinyInteger('votes');</code>	UNSIGNED TINYINT equivalent column.
<code>\$table-&gt;uuid('id');</code>	UUID equivalent column.
<code>\$table-&gt;year('birth_year');</code>	YEAR equivalent column.

এবার আমাদের বানানো এবং আগে থাকা(যদি থাকে) Migration গুলোকে দিয়ে আমরা আমাদের Database এ table গুলি বানিয়ে ফেলি। চলুন Terminal এ নিচের কমান্ডটি রান করি :

```
1 | php artisan migrate
```

আপনার CLI নিচের মত ফলাফল দেখাবে :

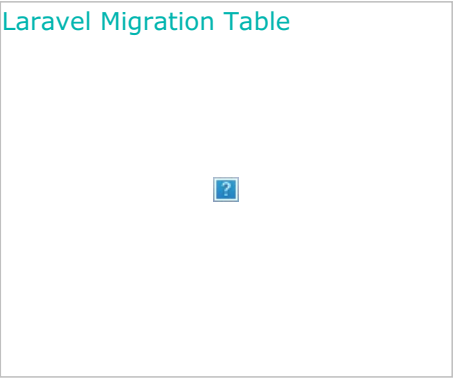
```
Command Prompt

C:\xampp\htdocs\w3programmers>php artisan migrate
Migration table created successfully.
Migrating: 2017_12_28_144409_create_students_table
Migrated: 2017_12_28_144409_create_students_table

C:\xampp\htdocs\w3programmers>
```

Laravel Migration Complete

উপরের screenshot দেখে বুঝা যাচ্ছে যে আমাদের Migration হয়ে গেছে। এখন আপনি আপনার Database টি দেখুন। এখানে আমাদের students table ছাড়াও আরও একটি টেবিল আছে যার নাম migrations. এটি লারাবেল নিজে ব্যবহার করে। নিচে আমাদের phpmyadmin screenshot result দেওয়া হলো:



Laravel Migration Table

মাইগ্রেশনটি rollback বা undo করা:

কখনো আমরা আমাদের database এর আগের অবস্থানে যেতে হতে পারে , ধরুন আপনি একটা Migration run করার পর আপনার মনে হলোঃ আপনার যেকোনো একটি column নামের বানান ভুল হয়েছে বা আরও একটি ফিল্ড add করতে হবে তখন আবার নতুন মাইগ্রেশন না লিখে rollback আপনি rollback করতে পারেন, তারপর আপনি আপনার মাইগ্রেশনটাকে প্রয়োজন মতো লিখে আবার মাইগ্রেন্ট কমান্ড রান করতে পারবেন। আর এর জন্য আপনাকে নিচের command টি রান করতে হবে :

```
1 | php artisan migrate:rollback
```

এখন আপনাকে ঠিক নিচের মত রেজাল্ট দেখাবে :

```
Command Prompt

C:\xampp\htdocs\w3programmers>php artisan migrate:rollback
Rolling back: 2017_12_28_144409_create_students_table
Rolled back: 2017_12_28_144409_create_students_table

C:\xampp\htdocs\w3programmers>
```

এখন আপনি আপনার table টা আর database এ দেখতে পারবেন না, অর্থাৎ , Larvel আপনার সর্বশেষ করা students table রিমুভ করে দিবে। এবং Laravel তার নিজস্ব table migrations টি রেখে দিবে।

**Note:**আপনার কোনও মাইগ্রেশনে যদি Rename অথবা Drop থাকে তাহলে উপরের command টি error দিবে। এই জন্য, doctrine/dbal নামক একটি প্যাকেজ আপনার প্রোজেক্ট এ যোগ করতে হবে। Terminal এ শুধু নিচের কমান্ডটি রান করুনঃ

```
1 | composer require doctrine/dbal
```

এটা আমাদের সব প্যাকেজ গুলোকে update, প্রয়োজনে add করে নিবে।

Migration ব্যবহার করে সব টেবিল গুলো মুছে ফেলা :

Laravel Framework এ Migration ব্যবহার করে সব table মুছে ফেলতে চাইলে আপনি নিচের কমান্ডটি রান করাই করতে পারেন :

```
1 | php artisan migrate:reset
```

এখন পুরা ডাটাবেজ ক্লিয়ার হয়ে যাবে এবং আবার যদি মাইগ্রেন্ট কমান্ড রান করাই তাহলে একে বারে ফ্রেশ একটি ডাটাবেজ পাবো।

একবারে সব টেবিল মুছে ফেলে নতুন করে তৈরি

একবারে সব টেবিল মুছে ফেলে নতুন করে তৈরি করতে চাইলে আপনি নিচের কমান্ডটি রান করাই করতে পারেন :

```
1 | php artisan migrate:refresh
```

এখন আপনাকে ঠিক নিচের মত রেজাল্ট দেখাবে :

```
Command Prompt

C:\xampp\htdocs\w3programmers>php artisan migrate:refresh
Rolling back: 2017_12_28_144409_create_students_table
Rolled back: 2017_12_28_144409_create_students_table
Migrating: 2017_12_28_144409_create_students_table
Migrated: 2017_12_28_144409_create_students_table

C:\xampp\htdocs\w3programmers>
```

laravel migration refresh

Masud Alam

Masud Alam



CEO at [TechBeeo Software Company](#)

আমি মাসুদ আলম, বাংলাদেশের ৩৬ তম Zend Certified Engineer । ২০০৯ সালে কম্পিউটার সাইন্স থেকে বেচেলর ডিগ্রী অর্জন করি। দীর্ঘ ৮ বছর আমি Winux Soft, SSL Wireless, IBCS-PRIMAX, Max Group, Canadian International Development Agency (CIDA), Care Bangladesh, World Vision, Hellen Keller সহ বিভিন্ন দেশি বিদেশী কোম্পানিতে ওয়েব ডেভেলপমেন্ট এবং সফটওয়্যার ডেভেলপমেন্ট এর উপর বিভিন্ন লিডিং পজিশন এ চাকরি এবং প্রজেক্ট লিড করি। বিশেষ ভাবে বাংলাদেশের ১০০ জন জেড সার্টিফাইড ইঞ্জিনিয়ার এর মধ্যে ৫২ জন ই আমার হাতে জেড সার্টিফাইড হয়েছে। বর্তমানে TechBeeo Software Company তে সিইও হিসাবে আছি । পাশাপাশি w3programmers ট্রেনিং ইনস্টিটিউট এ PHP এর উপর Professional এবং Advance Zend Certified PHP -7 Engineering কোর্স করাই। আর w3programmers.com সাইট টি আমার।

Tagged [laravel 5 tutorial for beginners](#), [Laravel 5.5](#), [laravel 5.6 tutorial](#), [laravel bangla blog](#), [laravel bangla book](#), [laravel bangla pdf book](#), [laravel bangla tutorial pdf](#), [laravel create database](#), [laravel delete migration](#), [Laravel Framework](#), [laravel make model](#), [laravel migration add column](#), [laravel migration change column type](#), [Laravel Migrations](#), [laravel mvc](#), [laravel project example](#), [laravel rollback specific migration](#), [laravel run specific migration](#), [laravel seeder](#), [laravel tutorial](#), [laravel tutorial for beginners step by step bangla](#), [laravel tutorial pdf](#), [laravel tutorial step by step](#), [laravel tutorial w3schools](#), [laravel tutorial youtube](#), [Searches related to laravel tutorial](#)

## Post navigation

[← Previous](#)

[Next →](#)

2 comments to “বাংলায় PHP LARAVEL FRAMEWORK পর্ব-৮: LARAVEL Migrations”

- Raju**  
[August 10, 2018 at 6:55 pm](#)

Nice...and good post, thanks sir.

- Anupam Roy**  
[May 22, 2019 at 9:56 am](#)

‘students’ name table na hoye keno ‘user’ name table hocee ?

## Leave a Reply

You must be [logged in](#) to post a comment.

Search for:

Video Player



# raptor not found.



## Categories

00:00 [Computer Networking](#)

00:00 [CSS](#)

01:04:59 [Git, gitHub, gitLab and Bitbucket](#)

• [HTML](#)  
Use Up/Down Arrow keys to increase or decrease volume.

• [JavaScript](#)

• [Laravel E-commerce Project](#)

- [OAuth](#)
- [PHP Arrays](#)
- [PHP Basic](#)
- [PHP Control Structures](#)
- [PHP Data Format & Types](#)
- [PHP Functions](#)
- [PHP Laravel Framework](#)
- [PHP MySQL](#)
- [PHP Object Oriented Programming](#)
- [PHP Operators](#)
- [PHP Socket Programming](#)
- [PHP Strings & Patterns](#)
- [PHP Variable & Constant](#)
- [PHP Web Features](#)
- [Python Basics](#)
- [Python Data Types](#)
- [VueJS](#)
- [WordPress Plugin Development](#)
- [Wordpress Theme Development](#)

## Join Our Group

**W3programmers**  
Facebook Group · 6,155 members

w3programmers is the Official group of the [www.w3programmers.com](http://www.w3programmers.com) website. Here you can discuss everything about Web Development & Technologies such...

## Like Us On Facebook

# raptor not found.



Copyright © 2021 w3programmers.com Bangla