Contents

[**SQL Query to create Database** 2](#_Toc45798619)

[**For basicPHP project** 2](#_Toc45798620)

[Description 2](#_Toc45798621)

[Adding Courses 2](#_Toc45798622)

[Editing Courses 3](#_Toc45798623)

[**For myapp project (Laravel)** 4](#_Toc45798624)

[Description 4](#_Toc45798625)

[Constraints 4](#_Toc45798626)

[Adding Courses 4](#_Toc45798627)

[Editing Courses 8](#_Toc45798628)

# **SQL Query to create Database**

CREATE TABLE courses(

id int(10) UNSIGNED NOT NULL,

course\_title varchar(100) NOT NULL UNIQUE,

course\_description text NOT NULL,

course\_deadline DATE NOT NULL,

CONSTRAINT pk\_courses PRIMARY KEY courses(id)

);

# **For basicPHP project**

## Description

* Validations have not been implemented here.
* OOP concepts are implemented
* Adding and updating courses done

## Adding Courses

1. Scenario 01 – Adding a valid course

POST Request: [*http://localhost:40/Sample/api/addCourse.php*](http://localhost:40/Sample/api/addCourse.php)

Request body:

*{*

*"course\_title": "Sofware Engineering",*

*"course\_description": "Software engineering is the discipline of designing, creating and maintaining software by applying technologies and practices from computer science, project management, engineering, application domains, interface design, digital assets management and other fields. ",*

*"course\_deadline": "2022-01-01"*

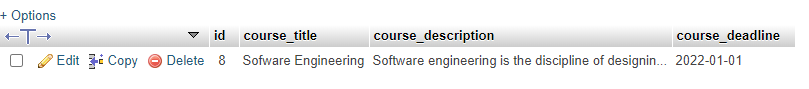
*}*

Response:

*{*

*"message": "Course added!"*

*}*



## Editing Courses

1. Scenario 01 – Updating a course

PUT Request: [*http://localhost:40/Sample/api/editCourse.php*](http://localhost:40/Sample/api/editCourse.php)

Request body:

*{*

*"id":8,*

*"course\_title": "Sofware Engineering",*

*"course\_description": "Software engineering is the discipline of designing, creating and maintaining software by applying technologies and practices from computer science, project management, engineering, application domains, interface design, digital assets management and other fields. ",*

*"course\_deadline": "1999=10-10"*

*}*

Response:

*{*

*"message": "Course updated!"*

*}*

\*changed the date



# **For myapp project (Laravel)**

## Description

* OOP concepts have not been implemented.
* All validations are done
* Updating and Adding Courses done.

## Constraints

* Title - required and minimum length can be 4
* Description – required, minimum length is 20, maximum length is 250
* Date – required and format check (yyyy-mm-dd)
* Duplicate entries not allowed for title

## Adding Courses

1. **Scenario 01 – Adding a course - successfully**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": "Business Management",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": "2020-10-09"*

*}*

Response:

*{*

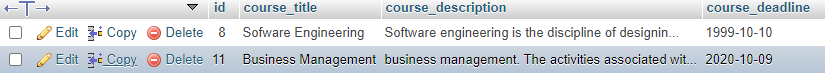
*"course\_title": "Business Management",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": "2020-10-09",*

*"id": 11*

*}*



1. **Scenario 02 – Adding a course – empty title**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": " ",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": "2020-10-09"*

*}*

Response:

*{*

*"course\_title": [*

*"The course title field is required."*

*]*

*}*

1. **Scenario 03 – Adding a course – empty description**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": "Business",*

*"course\_description": "",*

*"course\_deadline": "2020-10-09"*

*}*

Response:

*{*

*"course\_description": [*

*"The course description field is required."*

*]*

*}*

1. **Scenario 04– Adding a course – empty deadline**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": "Business",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": " "*

*}*

Response:

*{*

*"course\_deadline": [*

*"The course deadline field is required."*

*]*

*}*

1. **Scenario 05 – Adding a course – all invalid values**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": "Ah",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": "2020-10asdads9"*

*}*

Response:

*{*

*"course\_title": [*

*"The course title must be at least 4 characters."*

*],*

*"course\_description": [*

*"The course description may not be greater than 250 characters."*

*],*

*"course\_deadline": [*

*"The course deadline is not a valid date."*

*]*

*}*

1. **Scenario 06 – Adding a course – all empty**

POST Request: <http://127.0.0.1:8000/api/course>

Request body:

*{*

*"course\_title": "",*

*"course\_description": "",*

*"course\_deadline": " "*

*}*

Response:

*{*

*"course\_title": [*

*"The course title field is required."*

*],*

*"course\_description": [*

*"The course description field is required."*

*],*

*"course\_deadline": [*

*"The course deadline field is required."*

*]*

*}*

1. **Scenario 07 – Adding a course – check duplications**

POST Request: <http://127.0.0.1:8000/api/course>

Request Body:

*{*

*"course\_title": "Business Management",*

*"course\_description": "business management. The activities associated with running a company, such as controlling, leading, monitoring, organizing, and planning.",*

*"course\_deadline": "2020-10-09"*

*}*

Response:

*{*

*"course\_title": [*

*"The course title has already been taken."*

*]*

*}*

## Editing Courses

1. **Scenario 01: updating an existing record**

PUT Request: <http://127.0.0.1:8000/api/course/11>

Request Body:

*{*

*"course\_description": "This Course is no longer available ",*

*"course\_deadline": "1111-11-11"*

*}*

Response:

*{*

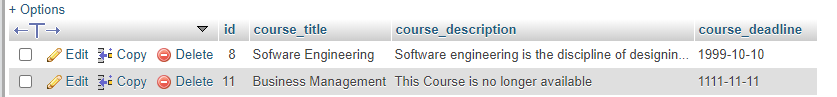
*"id": 11,*

*"course\_title": "Business Management",*

*"course\_description": "This Course is no longer available",*

*"course\_deadline": "1111-11-11"*

*}*



1. **Scenario 02 – updating a non-existing course**

PUT Request: <http://127.0.0.1:8000/api/course/100>

Request Body:

*{*

*"course\_description": "This Course is no longer available ",*

*"course\_deadline": "1111-11-11"*

*}*

Response:

*{*

*"message": "Record Not found!"*

*}*