**Student**

**1. Get all student**

**AP**I: localhost:3030/student/get/all

**Type: Rest**

**Method: Get**

**Access**: **:** admin

**Body :** N/A

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo:**

**WorkFlow: 1.** It will display all student data

**Teacher**

**1. Register a new teacher**

**AP**I: localhost:3030/teacher/registration

**Type: Rest**

**Method: POST**

**Access**: **:** Teacher, Admin

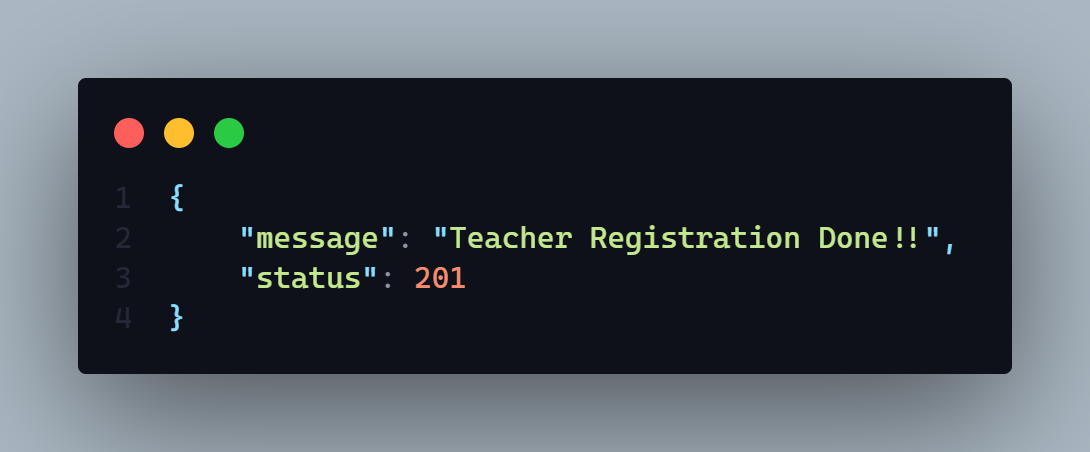
**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** It will create a new teacher

2. It will auto generate a unique teacher id and slug

3. In the time of registration if teacher will not want to upload a profile picture then it will auto upload a default picture in the server.

4. When teacher will input a password it will store in the database in hashed format.

**Courses**

**1. Create a new course**

**AP**I: localhost:3030/course/create

**Type: Rest**

**Method: POST**

**Access**: **:** Teacher

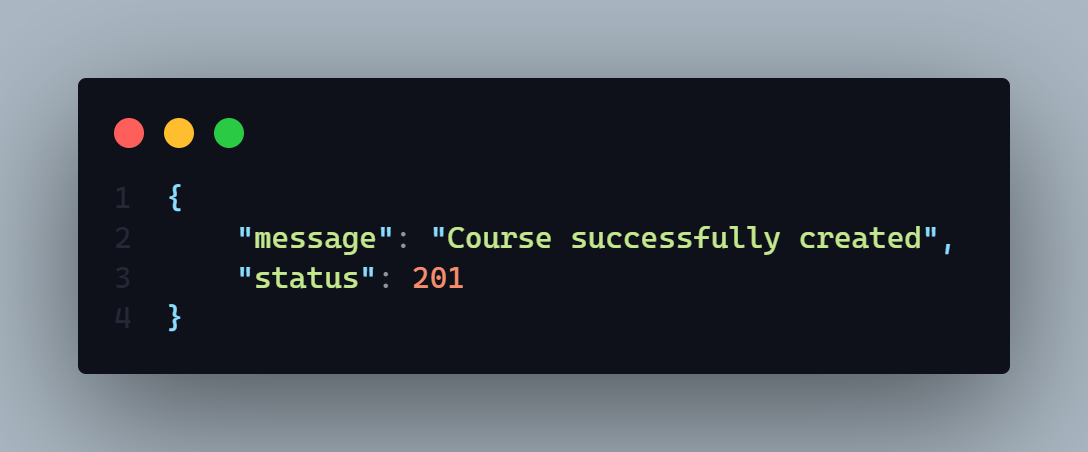
**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** Only Teacher can create a new course

2. When a new course has been created a random course Id and slug will be generate

3. course Id generation format will be => firstTwoWordOfCouresName – 5 random digit

4. slug generation format will be => coureseId\_duration\_creationTime

**2. Create a new MCQ question**

**AP**I: localhost:3030/course/create/quiz

**Type: Rest**

**Method: POST**

**Access**: **:** Teacher

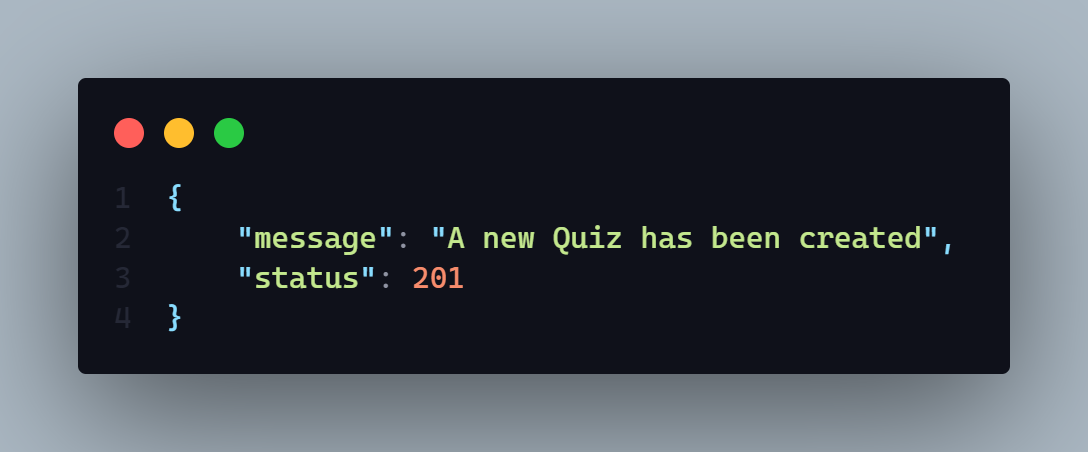
**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** This API will create a quiz with random quiz Id and slug.

2. Every quiz will contain many question with options including correct option name.

3. Quiz total mark will be count base on how many mark a single question have been contained.

4. To build a quiz it will create first a quiz then it will create some questions.

**Students**

**1. Register a new student**

**AP**I: localhost:3030/student/registration

**Type: Rest**

**Method: POST**

**Access**: **:** Student

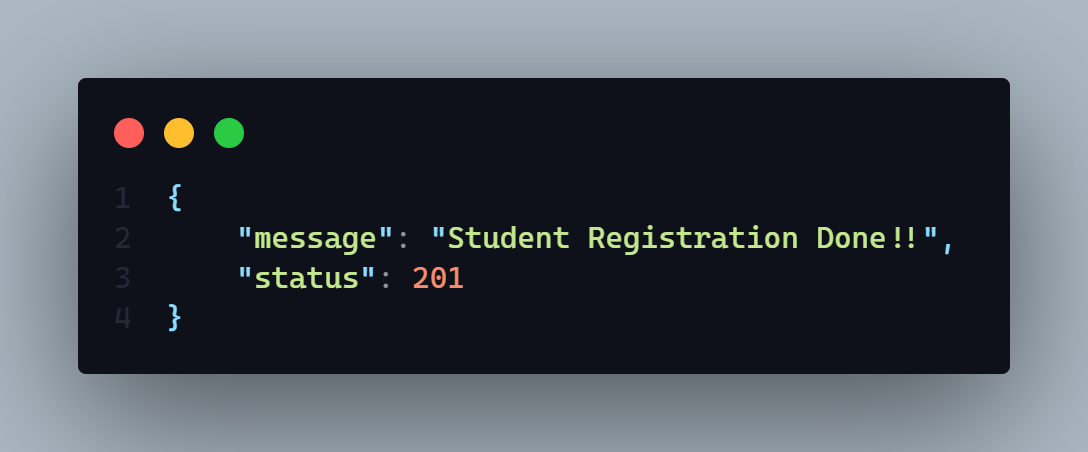
**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** It will create a new student

2. It will auto generate a unique student id and slug

3. In the time of registration if student will not want to upload a profile picture then it will auto upload a default picture in the server.

4. When student will input a password it will store in the database in hashed format.

2. **Enrolled a new Student**

**AP**I: localhost:3030/course/enrolled/course

**Type: Rest**

**Method: POST**

**Access**: **:** Student

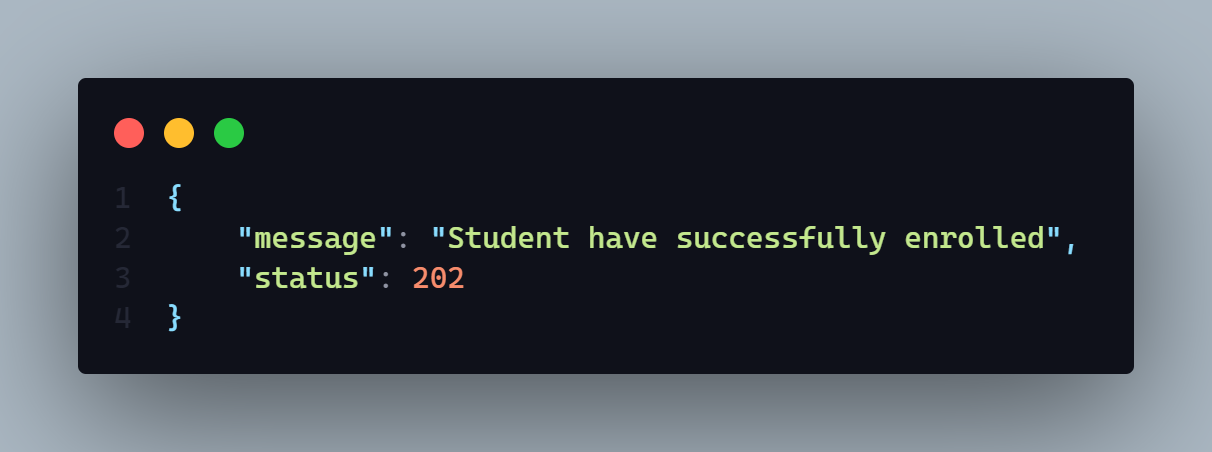
**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** It will enrolled a new student to existing course.

**3. Give a MCQ exam and get the result instantly**

**AP**I: localhost:3030/course/submit/quiz

**Type: Rest**

**Method: POST**

**Access**: **:** Student

**Body :**

****

**Params:** N/A

**Query:** N/A

**Response:**

****

**Demo: N/A**

**WorkFlow: 1.** When student give a MCQ exam it will take all option

2. It will match all given answer with the original answer and mark it and give the result instant.

.