

# *.NET Core Training*

*Building a RESTful API with  
ASP.NET Core 3*

Week 4 : 16<sup>th</sup> Feb – 22<sup>nd</sup> Feb

# AGENDA

1

Progress

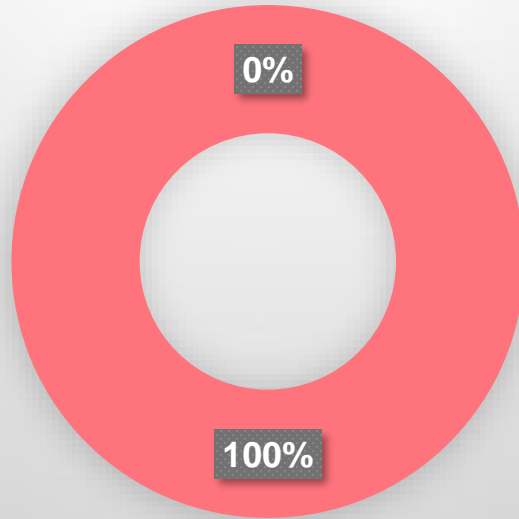
2

Demo

3

Learnings from the course

## Progress



- Building RESTful API (Completed)
- Building RESTful API (To Do)



DEMO

---

## Learnings from the Course



## Action Items

---

- Linked List Implementation
- REST API Implementation

# Course Learnings

---

## **HEAD :**

- The HEAD method is almost identical to GET, except without the response body.
- HEAD requests are useful for checking what a GET request will return before making a GET request

## **Idempotent Methods:**

An idempotent HTTP method is a HTTP method that can be called many times without different outcomes. It would not matter if the method is called only once, or ten times over. The result should be the same.

# Course Learnings

---

## Safe Method:

A method is considered safe when it doesn't change the resource representation.

HTTP Method	Idempotent	Safe
OPTIONS	yes	yes
GET	yes	yes
HEAD	yes	yes
PUT	yes	no
POST	no	no
DELETE	yes	no
PATCH	no	no



# Course Learnings

## **Validation and Validating Errors:**

Working with validation in a RESTful World :

- Define Validation Rules
- Check Validation Rules
- Report Validation Errors

Defining Validation Rules :

In ASP. Net Core rules are defined through,

- Data Annotations
- IValidatableObject
- Validate input , not Output

# Course Learnings

---

## **Validation and Validating Errors:**

Checking Validation Rules :

- ModelState

Example : `ModelState.IsValid()`

Reporting Validation Errors :

- Response status should be displayed (status code).
- Response body should contain validation errors.

# Course Learnings

---

## **JSON PATCH Operations:**

- ADD.
- REMOVE.
- REPLACE.
- COPY.
- MOVE.
- TEST.

*THANKYOU!*