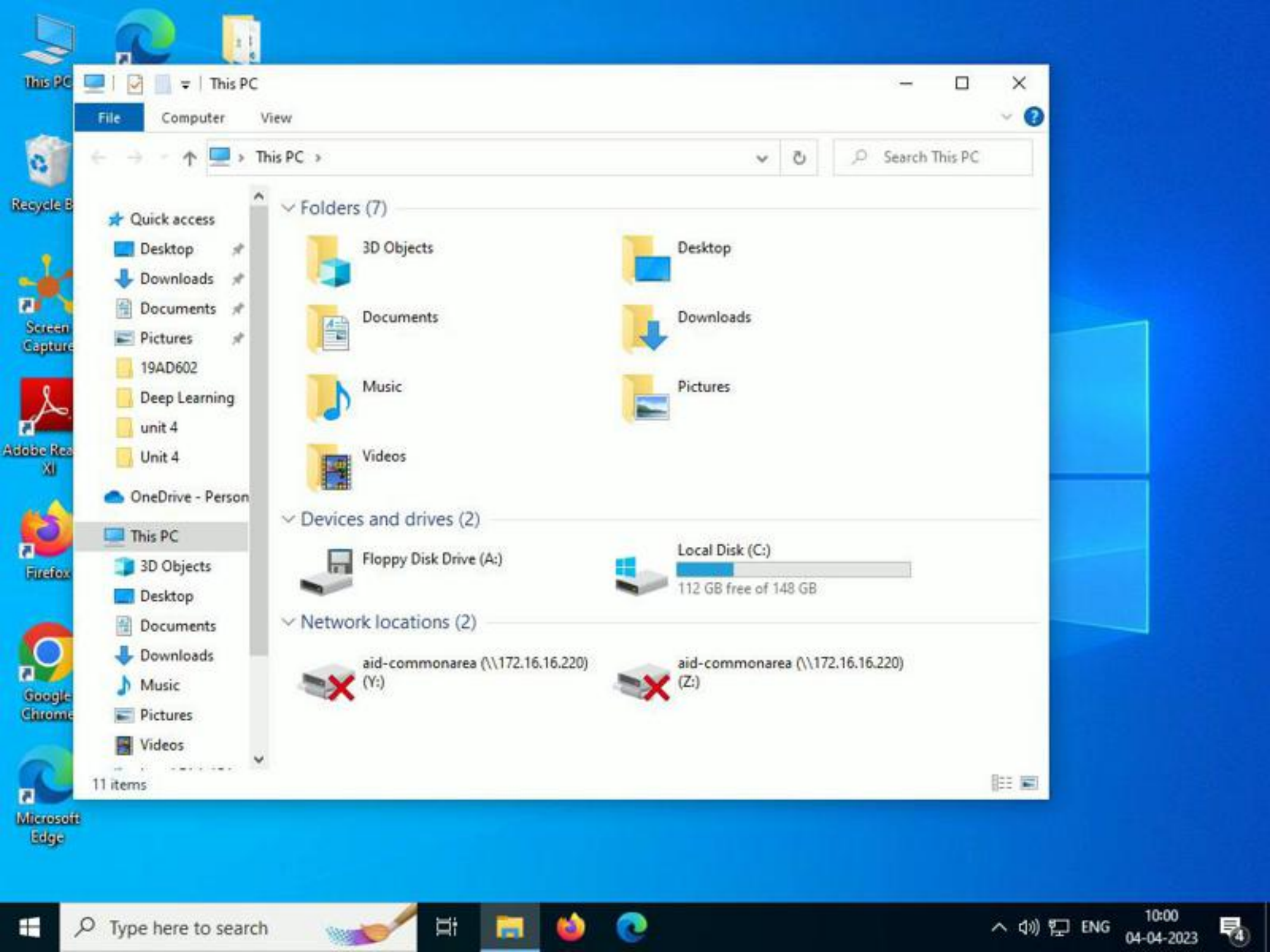
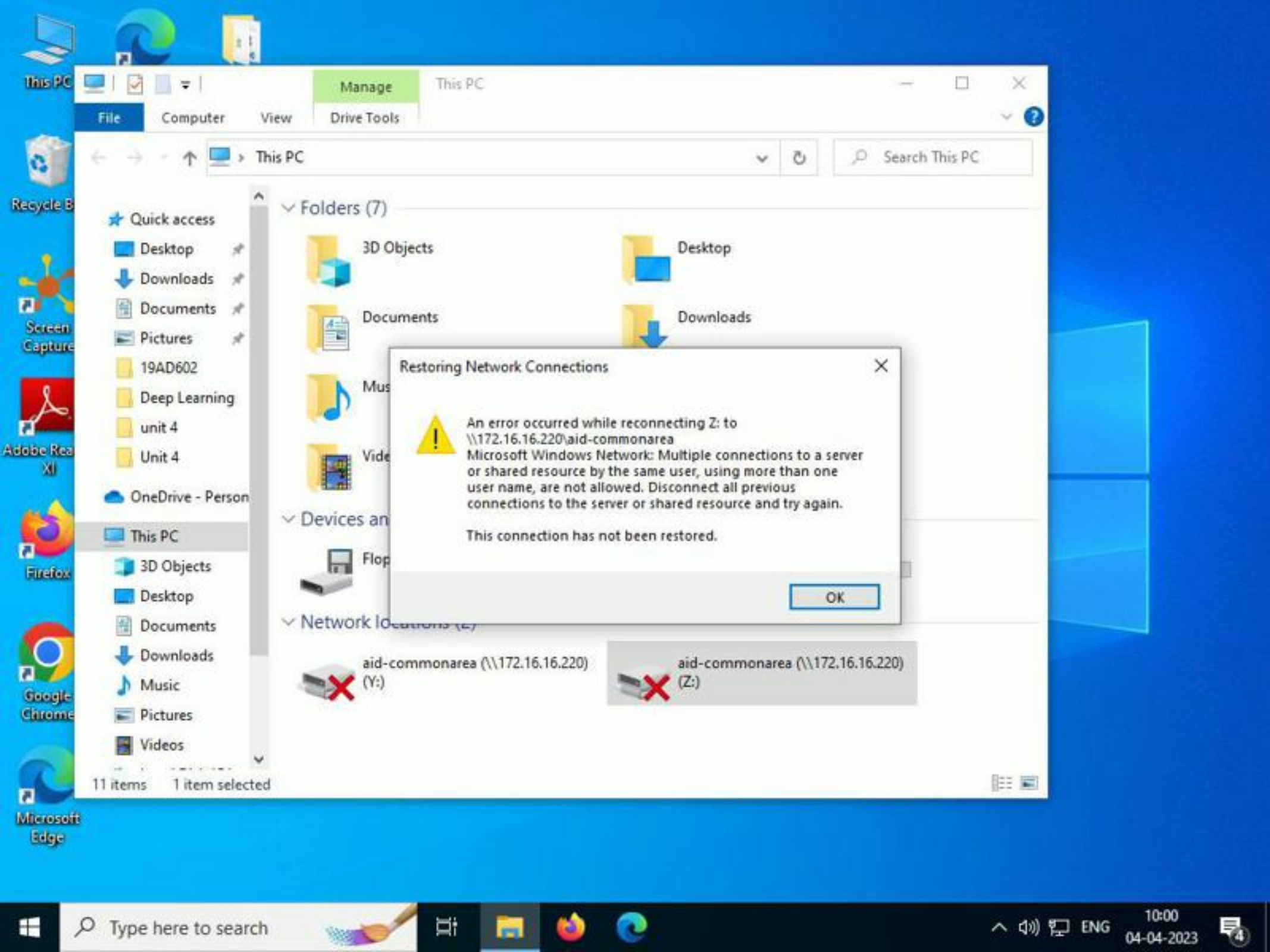
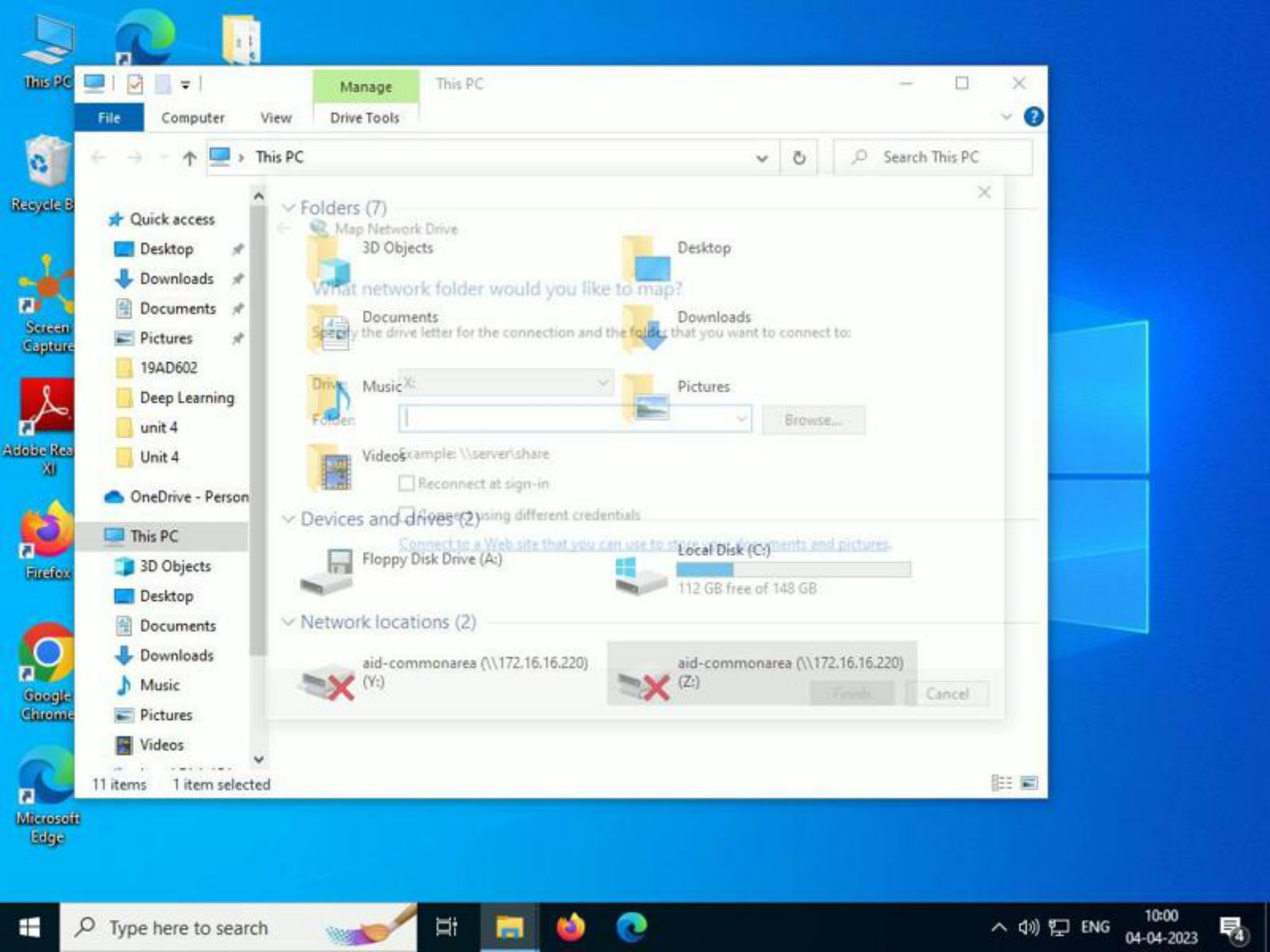
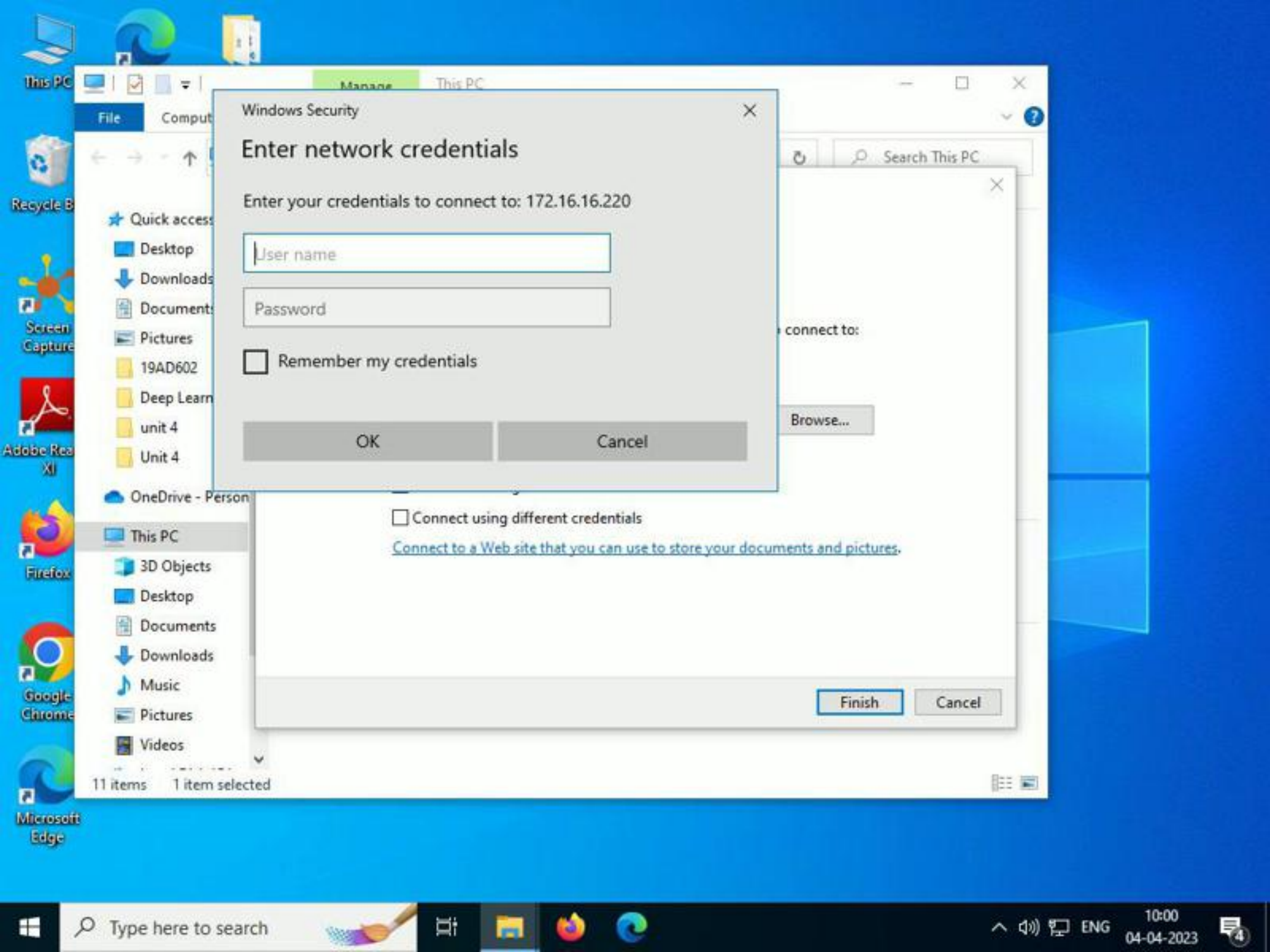
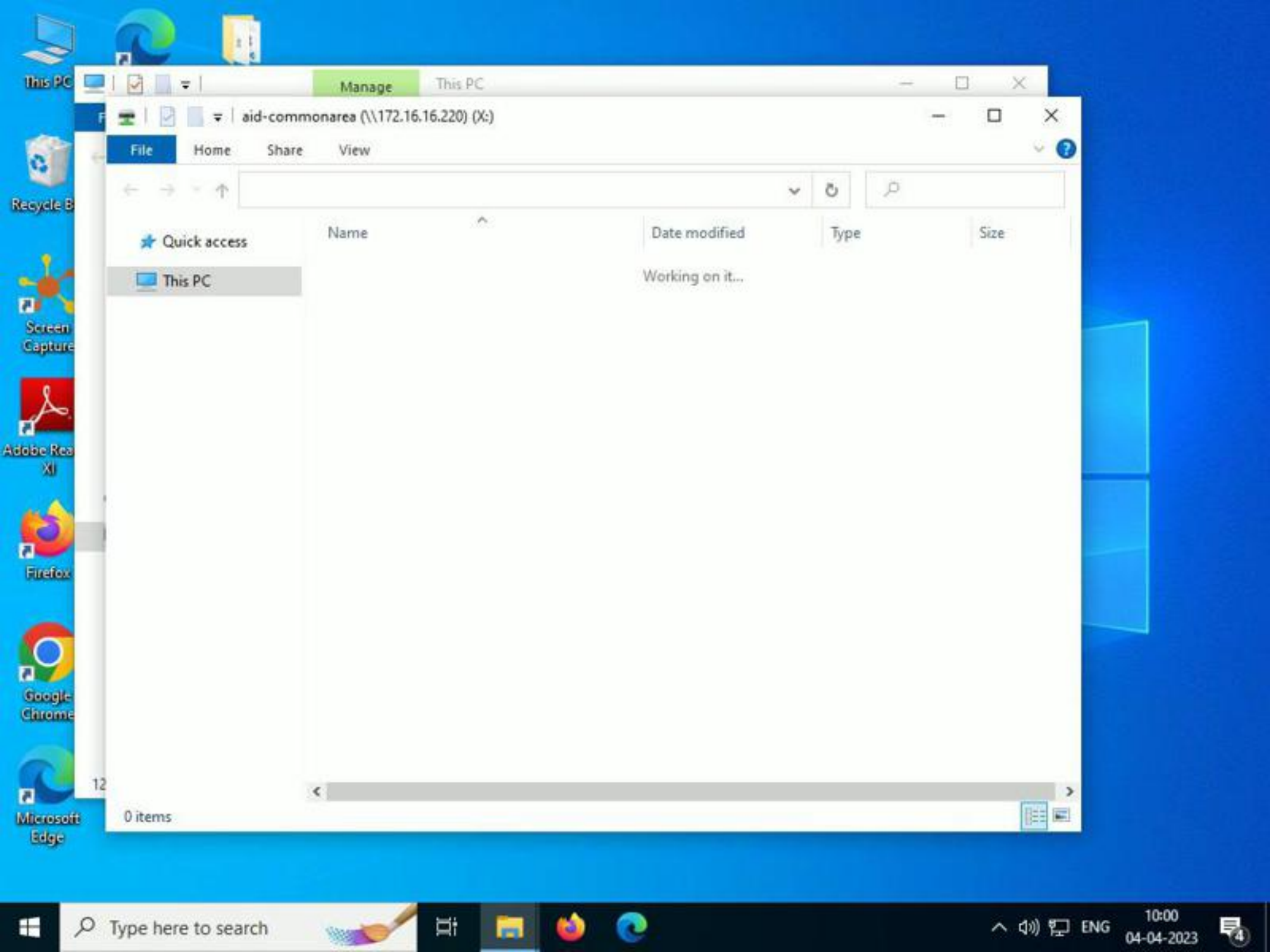
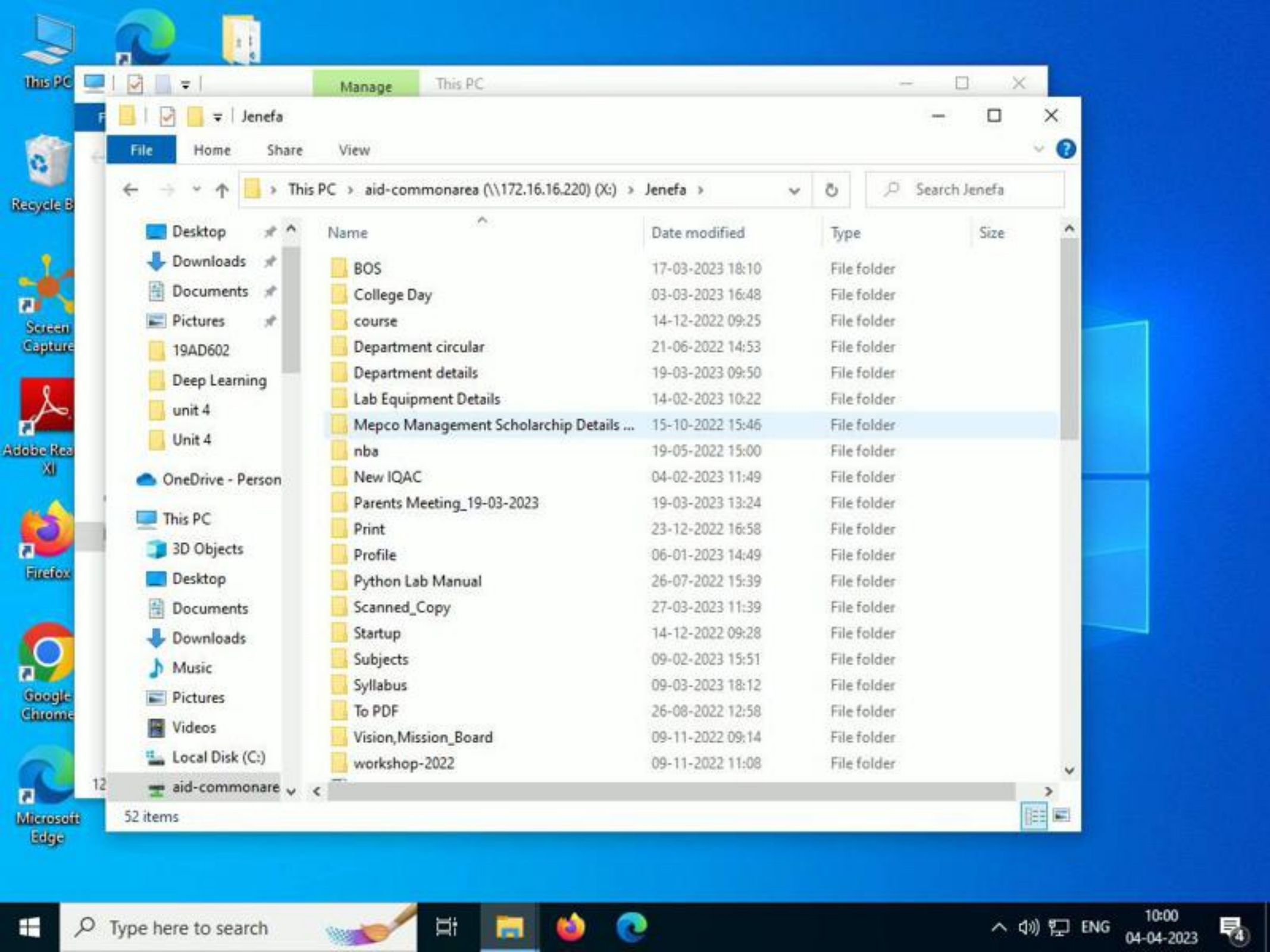

This PC
File
Computer
View
This PC
Search This PC
Quick access
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Pictures
Videos
Folders (7)
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Devices and drives (2)
Floppy Disk Drive (A:)
Local Disk (C:)
112 GB free of 148 GB
Network locations (2)
aid-commonarea (\\172.16.16.220) (Y:)
aid-commonarea (\\172.16.16.220) (Z:)
11 items
Type here to search
ENG
10:00
04-04-2023

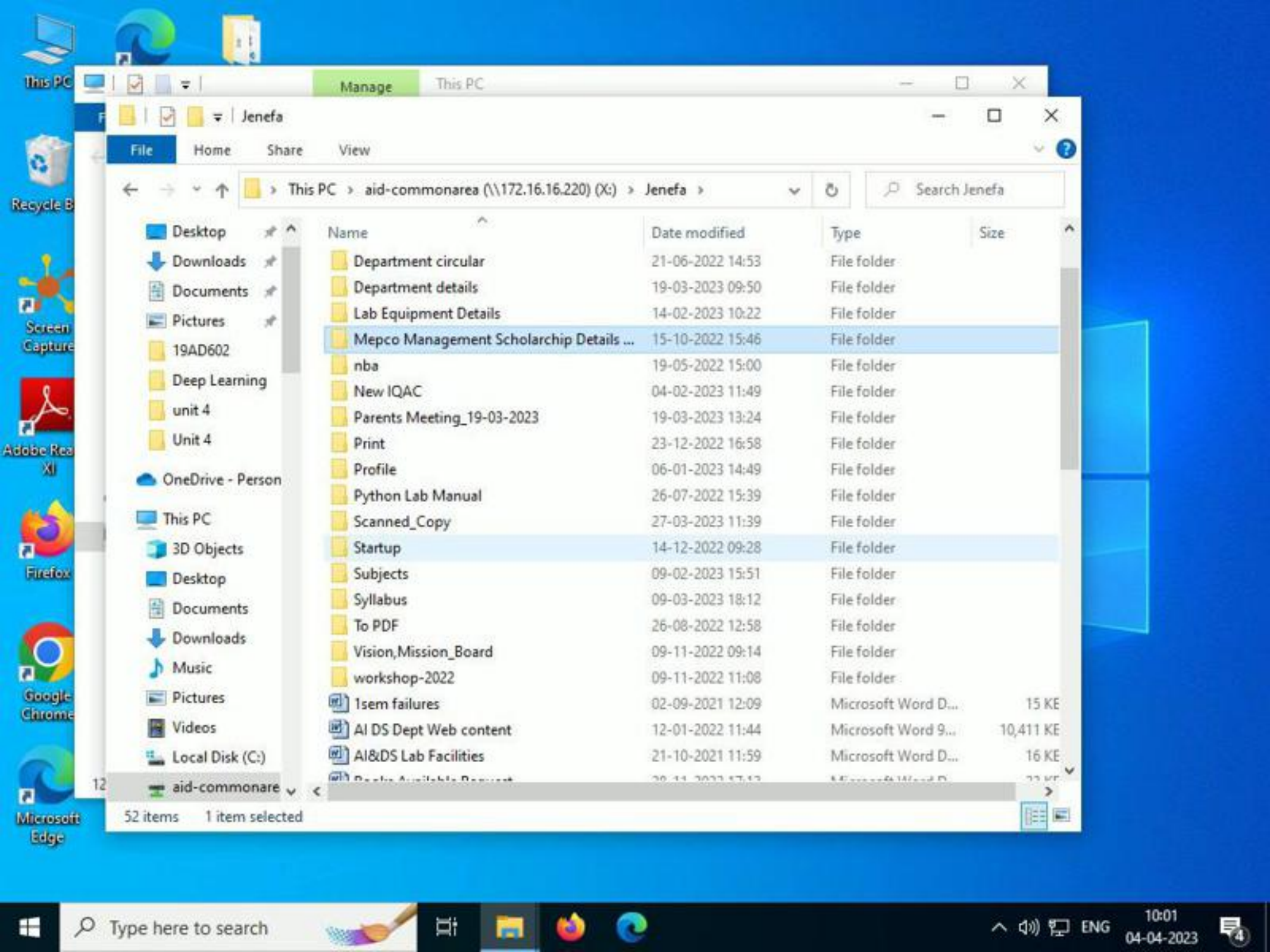

This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
File
Computer
View
Drive Tools
This PC
Search This PC
Quick access
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Pictures
Videos
Folders (7)
3D Objects
Desktop
Documents
Downloads
Restoring Network Connections
An error occurred while reconnecting Z: to \\172.16.16.220\aid-commonarea
Microsoft Windows Network: Multiple connections to a server or shared resource by the same user, using more than one user name, are not allowed. Disconnect all previous connections to the server or shared resource and try again.
This connection has not been restored.
OK
aid-commonarea (\\172.16.16.220) (Y:)
aid-commonarea (\\172.16.16.220) (Z:)
11 items
1 item selected
Type here to search
ENG
10:00
04-04-2023

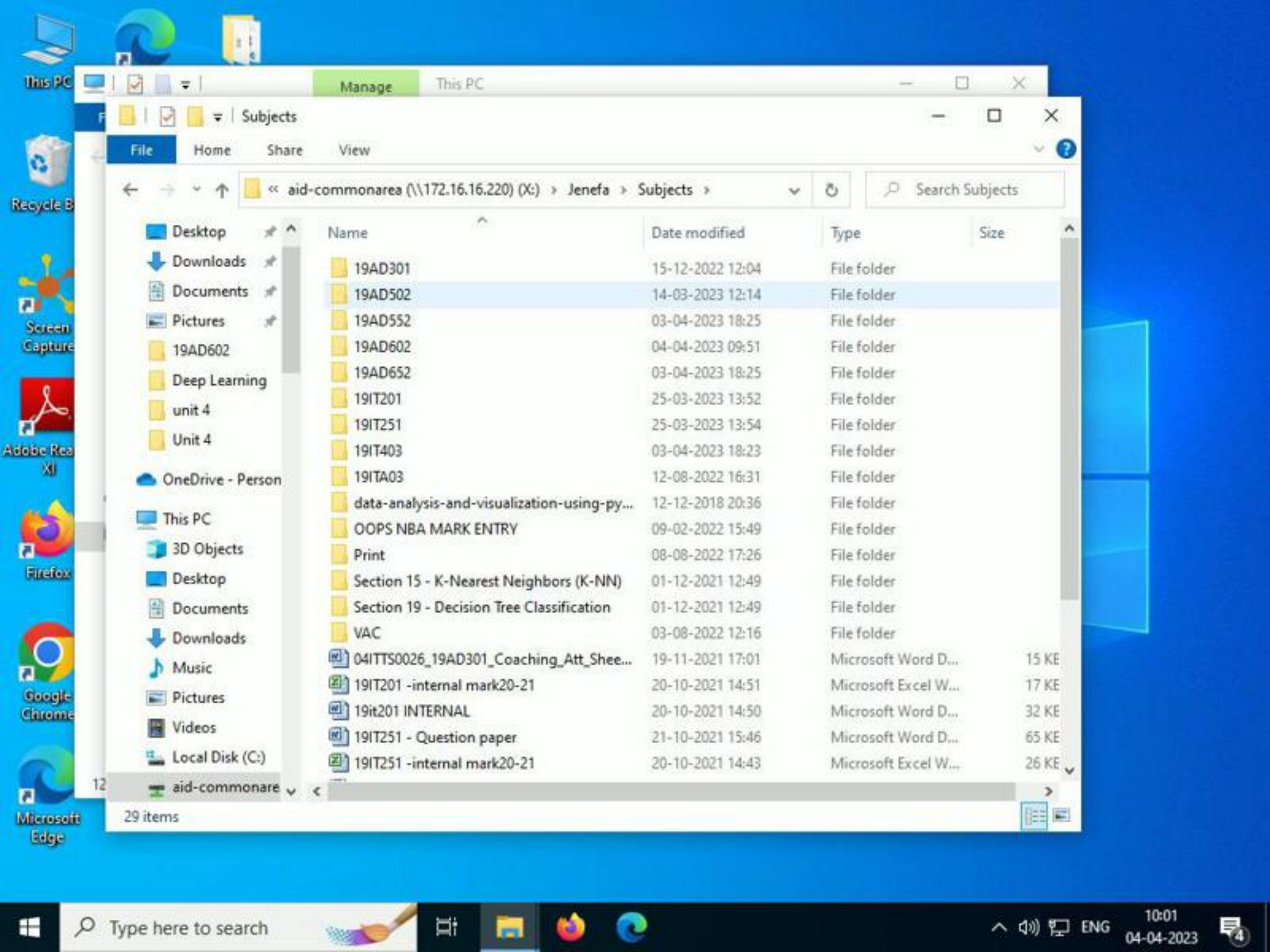

This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
File
Computer
View
Drive Tools
This PC
Search This PC
Quick access
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
11 items
1 item selected
Folders (7)
Map Network Drive
3D Objects
Desktop
What network folder would you like to map?
Documents
Downloads
Specify the drive letter for the connection and the folder that you want to connect to:
Drive:
Music
Pictures
Folder:
Browse...
Videos
Example: \\server\share
Reconnect at sign-in
Connect using different credentials
Connect to a Web site that you can use to store your documents and pictures.
Devices and drives (2)
Floppy Disk Drive (A:)
Local Disk (C:)
112 GB free of 148 GB
Network locations (2)
aid-commonarea (\\172.16.16.220) (Y:)
aid-commonarea (\\172.16.16.220) (Z:)
Finish
Cancel
Type here to search
ENG
10:00
04-04-2023

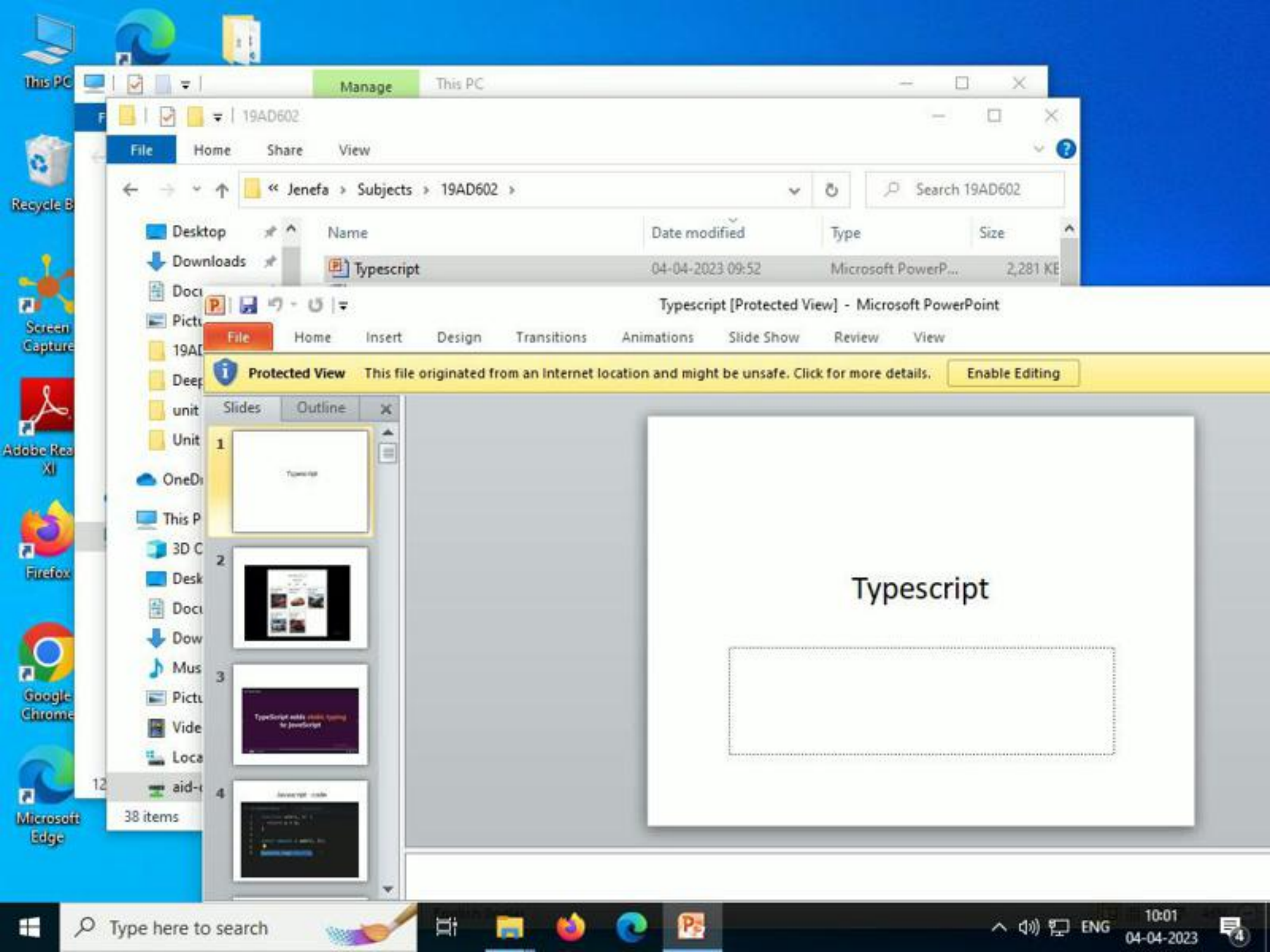

Windows Security
Enter network credentials
Enter your credentials to connect to: 172.16.16.220
User name
Password
Remember my credentials
OK
Cancel
Browse...
Connect using different credentials
Connect to a Web site that you can use to store your documents and pictures.
Finish
Cancel
11 items
1 item selected
Type here to search
ENG
10:00
04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
aid-commonarea (\\172.16.16.220) (X:)
File
Home
Share
View
Quick access
This PC
Name
Date modified
Type
Size
Working on it...
0 items
Type here to search
ENG
10:00
04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
Jenefa
File
Home
Share
View
This PC › aid-commonarea (\\172.16.16.220) (X:) › Jenefa ›
Search Jenefa
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
aid-commonare
Name
Date modified
Type
Size
BOS 17-03-2023 18:10 File folder
College Day 03-03-2023 16:48 File folder
course 14-12-2022 09:25 File folder
Department circular 21-06-2022 14:53 File folder
Department details 19-03-2023 09:50 File folder
Lab Equipment Details 14-02-2023 10:22 File folder
Mepco Management Scholarchip Details ... 15-10-2022 15:46 File folder
nba 19-05-2022 15:00 File folder
New IQAC 04-02-2023 11:49 File folder
Parents Meeting_19-03-2023 19-03-2023 13:24 File folder
Print 23-12-2022 16:58 File folder
Profile 06-01-2023 14:49 File folder
Python Lab Manual 26-07-2022 15:39 File folder
Scanned_Copy 27-03-2023 11:39 File folder
Startup 14-12-2022 09:28 File folder
Subjects 09-02-2023 15:51 File folder
Syllabus 09-03-2023 18:12 File folder
To PDF 26-08-2022 12:58 File folder
Vision,Mission_Board 09-11-2022 09:14 File folder
workshop-2022 09-11-2022 11:08 File folder
52 items
Type here to search
ENG
10:00
04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
Jenefa
File
Home
Share
View
This PC › aid-commonarea (\\172.16.16.220) (X:) › Jenefa ›
Search Jenefa
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
aid-commonare
Name
Date modified
Type
Size
Department circular 21-06-2022 14:53 File folder
Department details 19-03-2023 09:50 File folder
Lab Equipment Details 14-02-2023 10:22 File folder
Mepco Management Scholarchip Details ... 15-10-2022 15:46 File folder
nba 19-05-2022 15:00 File folder
New IQAC 04-02-2023 11:49 File folder
Parents Meeting_19-03-2023 19-03-2023 13:24 File folder
Print 23-12-2022 16:58 File folder
Profile 06-01-2023 14:49 File folder
Python Lab Manual 26-07-2022 15:39 File folder
Scanned_Copy 27-03-2023 11:39 File folder
Startup 14-12-2022 09:28 File folder
Subjects 09-02-2023 15:51 File folder
Syllabus 09-03-2023 18:12 File folder
To PDF 26-08-2022 12:58 File folder
Vision,Mission_Board 09-11-2022 09:14 File folder
workshop-2022 09-11-2022 11:08 File folder
1sem failures 02-09-2021 12:09 Microsoft Word D... 15 KE
AI DS Dept Web content 12-01-2022 11:44 Microsoft Word 9... 10,411 KE
AI&DS Lab Facilities 21-10-2021 11:59 Microsoft Word D... 16 KE
52 items
1 item selected
Type here to search
ENG
10:01
04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
Subjects
File
Home
Share
View
aid-commonarea (\\172.16.16.220) (X:) › Jenefa › Subjects ›
Search Subjects
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
aid-commonare
Name
Date modified
Type
Size
19AD301 15-12-2022 12:04 File folder
19AD502 14-03-2023 12:14 File folder
19AD552 03-04-2023 18:25 File folder
19AD602 04-04-2023 09:51 File folder
19AD652 03-04-2023 18:25 File folder
19IT201 25-03-2023 13:52 File folder
19IT251 25-03-2023 13:54 File folder
19IT403 03-04-2023 18:23 File folder
19ITA03 12-08-2022 16:31 File folder
data-analysis-and-visualization-using-py... 12-12-2018 20:36 File folder
OOPS NBA MARK ENTRY 09-02-2022 15:49 File folder
Print 08-08-2022 17:26 File folder
Section 15 - K-Nearest Neighbors (K-NN) 01-12-2021 12:49 File folder
Section 19 - Decision Tree Classification 01-12-2021 12:49 File folder
VAC 03-08-2022 12:16 File folder
04ITTS0026_19AD301_Coaching_Att_Shee... 19-11-2021 17:01 Microsoft Word D... 15 KE
19IT201 -internal mark20-21 20-10-2021 14:51 Microsoft Excel W... 17 KE
19it201 INTERNAL 20-10-2021 14:50 Microsoft Word D... 32 KE
19IT251 - Question paper 21-10-2021 15:46 Microsoft Word D... 65 KE
19IT251 -internal mark20-21 20-10-2021 14:43 Microsoft Excel W... 26 KE
29 items
Type here to search
ENG
10:01
04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
19AD602
File
Home
Share
View
Jenefa › Subjects › 19AD602 ›
Search 19AD602
Desktop
Downloads
Name
Date modified
Type
Size
Typescript
04-04-2023 09:52
Microsoft PowerP...
2,281 KE
38 items
Typescript [Protected View] - Microsoft PowerPoint
File
Home
Insert
Design
Transitions
Animations
Slide Show
Review
View
Protected View
This file originated from an Internet location and might be unsafe. Click for more details.
Enable Editing
Slides
Outline
Typescript
Type here to search
ENG
10:01
04-04-2023

Typescript [Protected View] - Microsoft PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View

Protected View This file originated from an Internet location and might be unsafe. Click for more details. Enable Editing

Slides Outline

1

2

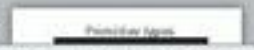
3

4

5

6

7



Typescript

Slide 1 of 67 "Office Theme" English (India) 76%

Type here to search ENG 10:01 04-04-2023






Getters and Setters

```
class Planet {
private _name: string = "No name set";
get name() {
return `This planet's name is '${this._name}'.`;
}
set name(inName: string) {
if (inName === "Pluto") {
this._name = "Not a planet";
} else {
this._name = inName;
}
}
}
let p: Planet = new Planet();
alert(p.name); // 'No name set'.
p.name = "Pluto";
alert(p.name); // 'Not a planet' (sorry, little guy!)
p.name = "Venus";
alert(p.name); // 'Venus'
```





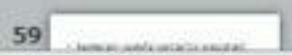

- TypeScript-specific syntax for important namespaces, the /// symbol.
- To use it, in the app.ts file,
- /// <reference path="file1.ts" />
- /// <reference path="file2.ts" />
- TypeScript, at compile time, will take care of bundling those files together.
- Once you have the code bundled
- import h = MyFirstNamespace.homeworld;
- alert(h) to see “Jakku.”







- namespaces can break into multiple files:
- // app.ts
- SomeNS.someFunc1();
- SomeNS.someFunc2();
- // file1.ts
- namespace SomeNS { export someFunc1() { } }
- // file2.ts
- namespace SomeNS { export someFund2() { } }



Namespaces and Modules

- Namespaces and modules are two concepts that help you to organize your TypeScript code by partitioning them in some logical fashion
- namespace MyFirstNamespace {
- export let homeworld = "Jakku";
- export function sayName() { alert("Rey"); };
- }
- alert(MyFirstNamespace.homeworld);
- MyFirstNamespace.sayName();
- namespace MyFirstNamespace {
- export class Jedi { }
- export interface RebelScum { }
- }

Extending Interfaces

- interface IPerson {
- firstName: string;
- }
- interface INinja extends IPerson {
- numberOfSwords: number;
- }
- let ninja = {} as INinja;
- ninja.firstName = "Ryuki";
- ninja.numberOfSwords = 2;

Namespaces and Modules

- Namespaces and modules are two concepts that help you to organize your TypeScript code by partitioning them in some logical fashion
- namespace MyFirstNamespace {
- export let homeworld = "Jakku";
- export function sayName() { alert("Rey"); };
- }
- alert(MyFirstNamespace.homeworld);
- MyFirstNamespace.sayName();
- namespace MyFirstNamespace {
- export class Jedi { }
- export interface RebelScum { }
- }

Namespaces and Modules

- Namespaces and modules are two concepts that help you to organize your TypeScript code by partitioning them in some logical fashion
- namespace MyFirstNamespace {
- export let homeworld = "Jakku";
- export function sayName() { alert("Rey"); };
- }
- alert(MyFirstNamespace.homeworld);
- MyFirstNamespace.sayName();
- namespace MyFirstNamespace {
- export class Jedi { }
- export interface RebelScum { }
- }

Namespaces

- JavaScript code produced from this TypeScript code:
- "use strict";
- var MyFirstNamespace;
- (function (MyFirstNamespace) {
- MyFirstNamespace.homeworld = "Jakku";
- function sayName() { alert("Rey"); }
- MyFirstNamespace.sayName = sayName;
- ;
- })(MyFirstNamespace || (MyFirstNamespace = {}));
- alert(MyFirstNamespace.homeworld);
- MyFirstNamespace.sayName();

- TypeScript uses the **IIFE (Immediately Invoked Function Expression)** pattern to keep the namespace's contents separate from everything else on the page when the code finally executes
- namespace MyFirstNamespace {
- export let homeworld = "Jakku";
- }
- const homeworld = "Coruscant";
- alert(MyFirstNamespace.homeworld); // Jakku
- alert(homeworld); // Coruscant

- namespaces can break into multiple files:
- // app.ts
- SomeNS.someFunc1();
- SomeNS.someFunc2();
- // file1.ts
- namespace SomeNS { export someFunc1() { } }
- // file2.ts
- namespace SomeNS { export someFund2() { } }

- To use it import both of the resultant .js files (file1.js and file2.js) in the HTML file
- Also need to import app.js
- Instead of having to import multiple .js files in the HTML document, you can instead have tsc bundle them for you:
- **tsc --outFile main.js file1.ts file2.ts app.ts**
- When bundling like this, you must be aware that order can matter. The files are concatenated in the order you provide

- TypeScript-specific syntax for important namespaces, the /// symbol.
- To use it, in the app.ts file,
- /// <reference path="file1.ts" />
- /// <reference path="file2.ts" />
- TypeScript, at compile time, will take care of bundling those files together.
- Once you have the code bundled
- import h = MyFirstNamespace.homeworld;
- alert(h) to see “Jakku.”

Nested namespace

- namespace SomeNS {
- export namespace InnerNS {
- export someFunc() { }
- }
- }
- SomeNS.InnerNS.someFunc();
- To use it:
- import sf = SomeNS.InnerNS.someFunc;
- sf();

- **namespaces** are more lightweight and are purely about **code organization**
- **classes** are about ***things***
- ***interfaces*** *are about* ***contracts***

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

Modules

- A module is defined as any TypeScript source file that contains one or more import or export statements
- // Variable
- export let captain = "Picard";
- // Interface
- export interface CaptainChecker {
- isGreat(inName: string): boolean;
- }
- // Function
- export function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- // Class
- export class Ship {
- const name = "Enterprise";
- }
- // Type alias
- export type cap = captain;

- Import { addFirst } from "./MyModule“
- execute it like any other function:
- addFirst("Riker"); // Wrong last name, but not the point!
- Alternatively, you could write your module like so:
- function addFirst(inLast: string): string {
- return "Jean Luc " + inLast;
- }
- export addFirst;

- export { addFirst as myAddFirstFunc }
- If you want to import an entire module, there is a handy shortcut for that:
- import * as TheModule from "./MyModule“
- export default let captain = "Picard";
- What that does for you is that now your import can be this:
- import cap from "./MyModule"

- export { addFirst as myAddFirstFunc }
- If you want to import an entire module, there is a handy shortcut for that:
- import * as TheModule from "./MyModule“
- export default let captain = "Picard";
- What that does for you is that now your import can be this:
- import cap from "./MyModule"

Typescript [Protected View] - Microsoft PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View

Protected View This file originated from an Internet location and might be unsafe. Click for more details. Enable Editing

Slides Outline

62

63

64

65

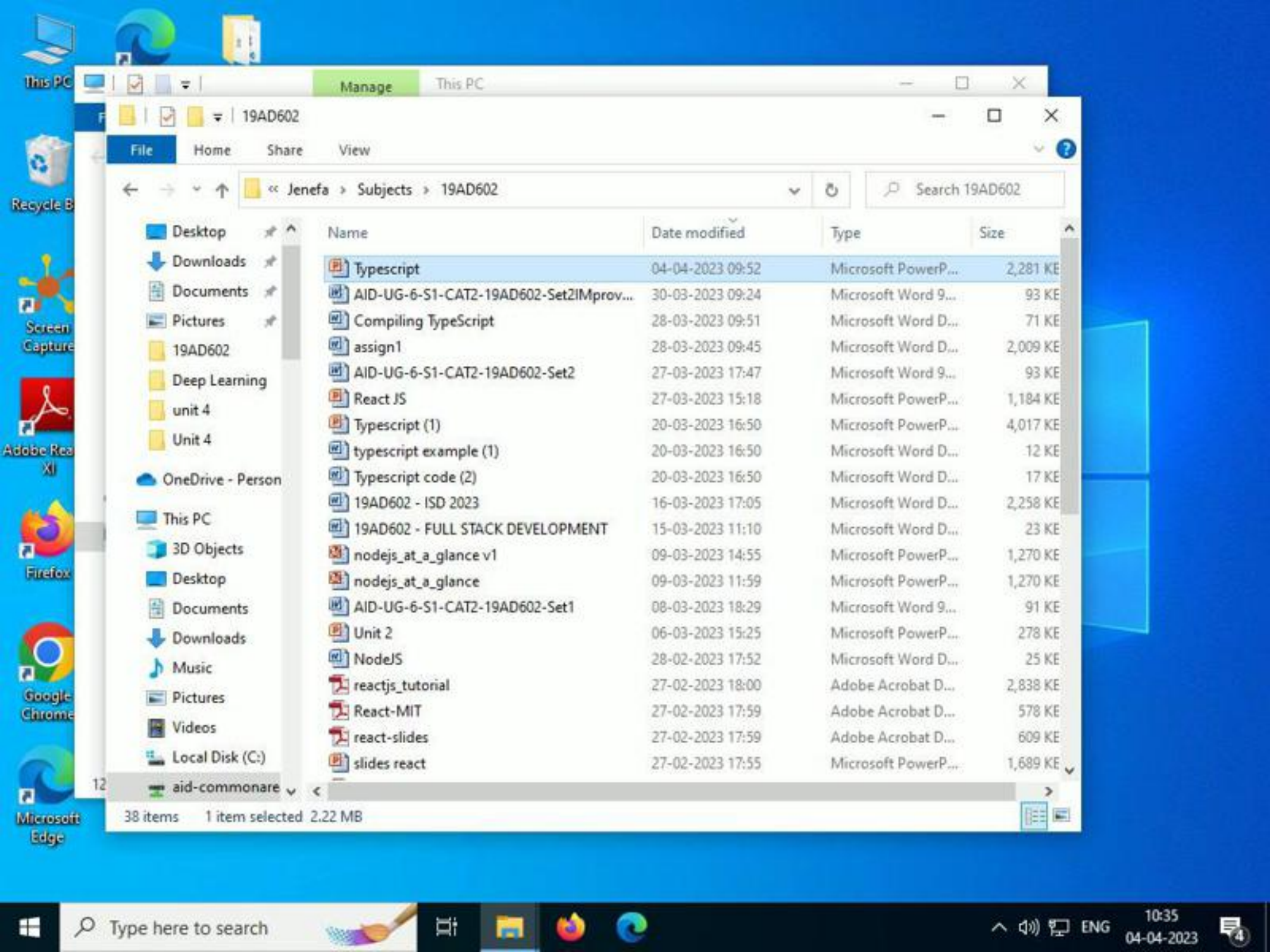
66

67

Slide 66 of 67 "Office Theme" English (India) 76%

Type here to search

ENG 10:35 04-04-2023


This PC
Recycle B
Screen Capture
Adobe Rea XI
Firefox
Google Chrome
Microsoft Edge
Manage
This PC
19AD602
File
Home
Share
View
Jenefa › Subjects › 19AD602
Search 19AD602
Desktop
Downloads
Documents
Pictures
19AD602
Deep Learning
unit 4
Unit 4
OneDrive - Person
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
aid-commonare
Name
Date modified
Type
Size
Typescript 04-04-2023 09:52 Microsoft PowerP... 2,281 KE
AID-UG-6-S1-CAT2-19AD602-Set2IMprov... 30-03-2023 09:24 Microsoft Word 9... 93 KE
Compiling TypeScript 28-03-2023 09:51 Microsoft Word D... 71 KE
assign1 28-03-2023 09:45 Microsoft Word D... 2,009 KE
AID-UG-6-S1-CAT2-19AD602-Set2 27-03-2023 17:47 Microsoft Word 9... 93 KE
React JS 27-03-2023 15:18 Microsoft PowerP... 1,184 KE
Typescript (1) 20-03-2023 16:50 Microsoft PowerP... 4,017 KE
typescript example (1) 20-03-2023 16:50 Microsoft Word D... 12 KE
Typescript code (2) 20-03-2023 16:50 Microsoft Word D... 17 KE
19AD602 - ISD 2023 16-03-2023 17:05 Microsoft Word D... 2,258 KE
19AD602 - FULL STACK DEVELOPMENT 15-03-2023 11:10 Microsoft Word D... 23 KE
nodejs_at_a_glance v1 09-03-2023 14:55 Microsoft PowerP... 1,270 KE
nodejs_at_a_glance 09-03-2023 11:59 Microsoft PowerP... 1,270 KE
AID-UG-6-S1-CAT2-19AD602-Set1 08-03-2023 18:29 Microsoft Word 9... 91 KE
Unit 2 06-03-2023 15:25 Microsoft PowerP... 278 KE
NodeJS 28-02-2023 17:52 Microsoft Word D... 25 KE
reactjs_tutorial 27-02-2023 18:00 Adobe Acrobat D... 2,838 KE
React-MIT 27-02-2023 17:59 Adobe Acrobat D... 578 KE
react-slides 27-02-2023 17:59 Adobe Acrobat D... 609 KE
slides react 27-02-2023 17:55 Microsoft PowerP... 1,689 KE
38 items 1 item selected 2.22 MB
Type here to search
ENG
10:35
04-04-2023