# Assignment #1 – code explanation

1. What is the output of the code below

static async Task Main(string[] args)

{

Console.WriteLine("Async action program");

await Task.Run(() =>

{

Task.Delay(1000);

Console.Out.WriteLine("delayed message part 1");

});

Task.Run(async () =>

{

await Task.Delay(1000);

Console.Out.WriteLine("delayed message part 2");

});

Console.WriteLine("Post task message");

await Task.Run(() => { Console.Out.WriteLine("async action completed"); });

}

1. Explain what’s wrong with the code above
2. Fix the code above (if needed in order to log all sysout)

Please explain all changes in the code

**#1**

**1.**

static async Task Main(string[] args)

{

Console.WriteLine("Async action program"); //'Async action program'מדפיס הודעה זו למסך:

await Task.Run(() => // Await הרצת משימה ראשונה עם

{

Task.Delay(1000); // לכן לא מחכים בפועל והקוד ממשיך מיד להדפסה אחריה Await כיון שאין שימוש ב-

Console.Out.WriteLine("delayed message part 1");// הקוד מדפיס את ההודעה למסך

});

Task.Run(async () => //שרצה באופן אסינכרוני ומאפשר להמשיך במקביל Awaitהרצת משימה שניה ללא

{

await Task.Delay(1000);// בתוך המשימה יש המתנה של שניה לפני ההדפסה

Console.Out.WriteLine("delayed message part 2");// מדפיס מיד אח"כ את ההודעה למסך

});

Console.WriteLine("Post task message");// הדפסת הודעה למסך אחרי המשימה

await Task.Run(() => { Console.Out.WriteLine("async action completed"); });  
הדפסה מיידית(משימה שלישית)//

}

**Output:**

**1-** Async action program

**2-** delayed message part 1

**3-** Post task message

**4-** async action completed

**5-** delayed message part 2 **//לאחר השהיה של שניה**

**2. \* 'task.delay' is not awaited in the first task**

Task.Delay(1000) is called but not awaited, this means the delay doesn't actually pause the task, and the subsequent Console.Out.WriteLine("delayed message part 1") executes immediately.

**\* The second Task.run is not awaited,**so the main method continues executing without waiting for this task to complete,  
that could lead to unexpected output if this line will be printed last.

**3.**

static async Task Main(string[] args)

{

Console.WriteLine("Async action program");

await Task.Run(() =>

{

**await Task.Delay(1000);// בתוך ההשהיה של המשימה הראשונהawait הוספת**

Console.Out.WriteLine("delayed message part 1");

});

**await Task.Run(async () => //לפני המשימה השניה כדי שימתינו להדפסה שלה await**

{

await Task.Delay(1000);

Console.Out.WriteLine("delayed message part 2");

});

Console.WriteLine("Post task message");

await Task.Run(() => { Console.Out.WriteLine("async action completed"); });

}

# Assignment #2 - User’s management API

Write an WEBAPI Controller with the following Methods

1. Create user
2. Delete user by UserId
3. Validate User by user and password

Guidance:

* Use DB Repository pattern
* Use Dependency injection pattern
* May Use in memory database / file / any other database
* User schema fields:

1. UserId
2. UserName
3. UserPassword

Please upload the source code to any Source control such as (github ,gitlab ,bitbucket etc)