

Name: Nidhi Singh

UID: 23BCS80101

Virtual Reality Tours

Code

TravelPlanner.cs

```
using UnityEngine;
```

```
using UnityEngine.UI;
```

```
using System.Collections.Generic;
```

```
public class TravelPlanner : MonoBehaviour
```

```
{
```

```
    [Header("UI Elements")]
```

```
    public Dropdown destinationDropdown; // Dropdown to select a  
destination
```

```
    public Text destinationDetailsText; // Text for showing details
```

```
    public InputField daysInputField; // Input field to enter the number of  
days
```

```
    public Text estimatedCostText; // Text to display the estimated cost
```

```
    public Button addToItineraryButton; // Button to add a destination to  
the itinerary
```

```
public Text itineraryText; // Text displaying the full itinerary

public Button finalizePlanButton; // Button to finalize the plan


[Header("Destination Data")]

public List<Destination> destinations = new List<Destination>(); //
List of available destinations

private List<string> itinerary = new List<string>(); // User's selected
destinations


private float costPerDay = 100.0f; // Average cost per day per
destination


void Start()
{
    PopulateDropdown();
    estimatedCostText.text = "Estimated Cost: $0";
    itineraryText.text = "Itinerary:\n";
}


void PopulateDropdown()
{
    // Populate the dropdown with destination names
```

```
destinationDropdown.ClearOptions();  
List<string> options = new List<string>();  
foreach (var destination in destinations)  
{  
    options.Add(destination.name);  
}  
destinationDropdown.AddOptions(options);  
}
```

```
public void OnDestinationSelected()  
{  
    // Update the details text based on the selected destination  
    int selectedIndex = destinationDropdown.value;  
    var selectedDestination = destinations[selectedIndex];  
    destinationDetailsText.text = $"Destination:  
{selectedDestination.name}\n" +  
        $"Description:  
{selectedDestination.description}\n" +  
        $"Average Cost per Day: ${costPerDay}";  
}
```

```

public void AddToItinerary()
{
    int selectedIndex = destinationDropdown.value;
    var selectedDestination = destinations[selectedIndex];

    if (!itinerary.Contains(selectedDestination.name))
    {
        itinerary.Add(selectedDestination.name);
        itineraryText.text += $"- {selectedDestination.name}\n";
    }

    UpdateEstimatedCost();
}

void UpdateEstimatedCost()
{
    // Calculate and display the estimated cost
    if (float.TryParse(daysInputField.text, out float days))
    {
        float totalCost = itinerary.Count * days * costPerDay;
        estimatedCostText.text = $"Estimated Cost: ${totalCost}";
    }
}

```

```
    }  
}  
  
public void FinalizePlan()  
{  
    // Finalize the travel plan (e.g., confirm booking)  
    Debug.Log("Travel Plan Finalized:\n" + string.Join(", ", itinerary));  
}  
}
```