HOTSPOT -

You are building a chatbot for a Microsoft Teams channel by using the Microsoft Bot Framework SDK. The chatbot will use the following code.

```
protected override async Task OnMembersAddedAsync(IList<ChannelAccount>
membersAdded, ITurnContext<IConversationUpdateActivity> turnContext,
CancellationToken cancellationToken)
{
    foreach (var member in membersAdded)
        if (member.Id != turnContext.Activity.Recipient.Id)
            await turnContext.SendActivityAsync($"Hi there - {member.Name}.
{WelcomeMessage}", cancellationToken: cancellationToken);
}
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Statements	Yes	No
`OnMembersAddedAsync` will be triggered when a user joins the conversation.	○	○
When a new user joins the conversation, the existing users in the conversation will see the chatbot greeting.	○	○
`OnMembersAddedAsync` will be initialized when a user sends a message.	○	○

Correct Answer:

Answer Area

Statements	Yes	No
`OnMembersAddedAsync` will be triggered when a user joins the conversation.	○	○
When a new user joins the conversation, the existing users in the conversation will see the chatbot greeting.	○	○
`OnMembersAddedAsync` will be initialized when a user sends a message.	○	○

Box 1: Yes -

The ActivityHandler.OnMembersAddedAsync method overrides this in a derived class to provide logic for when members other than the bot join the conversation, such as your bot's welcome logic.

Box 2: Yes -

membersAdded is a list of all the members added to the conversation, as described by the conversation update activity.

Box 3: No -

Reference:

https://docs.microsoft.com/en-us/dotnet/api/microsoft.bot.builder.activityhandler.onmembersaddedasync?view=botbuilder-dotnet-stable

Question #9 *Topic 5*

HOTSPOT -

You are reviewing the design of a chatbot. The chatbot includes a language generation file that contains the following fragment.

Greet(user)

- ${Greeting()}, ${user.name}

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Statements	Yes	No
`${user.name}` retrieves the user name by using a prompt.	○	○
`Greet()` is the name of the language generation template.	○	○
`${Greeting()}` is a reference to a template in the language generation file.	○	○

Correct Answer:

Answer Area

Statements	Yes	No
`${user.name}` retrieves the user name by using a prompt.	○	●
`Greet()` is the name of the language generation template.	○	●
`${Greeting()}` is a reference to a template in the language generation file.	●	○

Box 1: No -

Example: Greet a user whose name is stored in `user.name`

- ${ welcomeUser(user.name) }

Example: Greet a user whose name you don't know:

- ${ welcomeUser() }

Box 2: No -

Greet(User) is a Send a response action.

Box 3: Yes -

Reference:

https://docs.microsoft.com/en-us/composer/how-to-ask-for-user-input

Question #10 *Topic 5*

HOTSPOT -

You are building a chatbot by using the Microsoft Bot Framework SDK.

You use an object named UserProfile to store user profile information and an object named ConversationData to store information related to a conversation.

You create the following state accessors to store both objects in state. var userStateAccessors = _userState.CreateProperty<UserProfile> (nameof(UserProfile)); var conversationStateAccessors = _conversationState.CreateProperty<ConversationData>(nameof(ConversationData));

The state storage mechanism is set to Memory Storage.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Statements	Yes	No
The code will create and maintain the `UserProfile` object in the underlying storage layer.	○	○
The code will create and maintain the `ConversationData` object in the underlying storage layer.	○	○
The `UserProfile` and `ConversationData` objects will persist when the Bot Framework runtime terminates.	○	○

Correct Answer:

Answer Area

Statements	Yes	No
The code will create and maintain the `UserProfile` object in the underlying storage layer.	○	○
The code will create and maintain the `ConversationData` object in the underlying storage layer.	○	○
The `UserProfile` and `ConversationData` objects will persist when the Bot Framework runtime terminates.	○	○

Box 1: Yes -

You create property accessors using the CreateProperty method that provides a handle to the BotState object. Each state property accessor allows you to get or set the value of the associated state property.

Box 2: Yes -

Box 3: No -

Before you exit the turn handler, you use the state management objects' SaveChangesAsync() method to write all state changes back to storage.

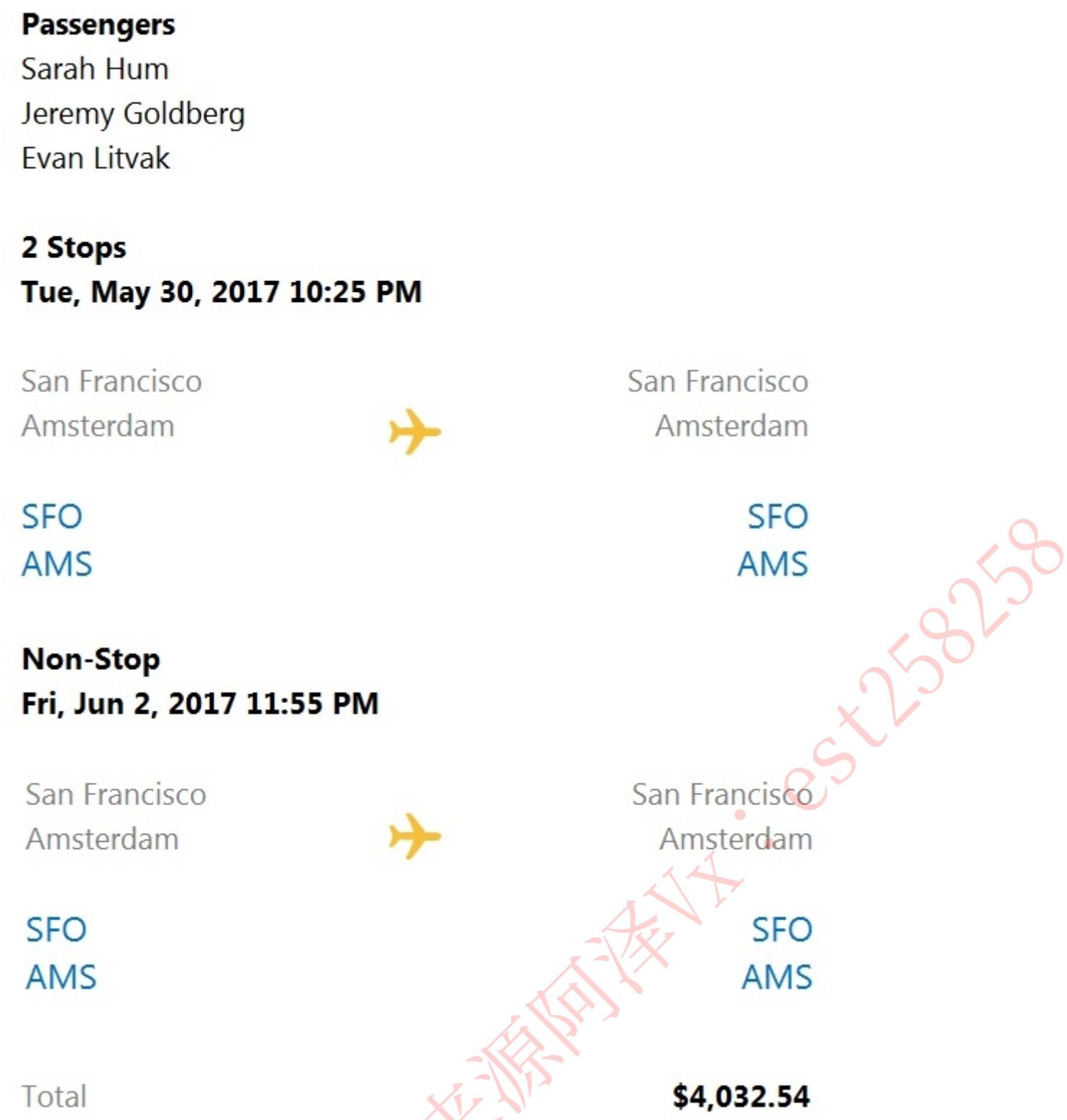
Reference:

https://docs.microsoft.com/en-us/azure/bot-service/bot-builder-howto-v4-state

Question #11 *Topic 5*

HOTSPOT -

You are building a chatbot that will provide information to users as shown in the following exhibit.



Use the drop-down menus to select the answer choice that completes each statement based on the information presented in the graphic.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

The chatbot is showing **[answer choice]**. ▼

- an Adaptive Card
- a Hero Card
- a Thumbnail Card

The card includes **[answer choice]**. ▼

- an action set
- an image
- an image group
- media

Correct Answer:

Answer Area

The chatbot is showing **[answer choice]**.

- an Adaptive Card
- a Hero Card
- a Thumbnail Card

The card includes **[answer choice]**.

- an action set
- an image
- an image group
- media

Box 1: A Thumbnail card -

A Thumbnail card typically contains a single thumbnail image, some short text, and one or more buttons.

Incorrect Answers:

☞ an Adaptive card is highly customizable card that can contain any combination of text, speech, images, buttons, and input fields.

☞ a Hero card typically contains a single large image, one or more buttons, and a small amount of text.

Box 2: an image -

Reference:

https://docs.microsoft.com/en-us/microsoftteams/platform/task-modules-and-cards/cards/cards-reference

Question #12 *Topic 5*

HOTSPOT -

You are building a bot and that will use Language Understanding.

You have a LUDown file that contains the following content.

```
## Confirm
- confirm
- ok
- yes

## ExtractName
- call me steve !
- i am anna
- (i'm|i am) {@PersonName.Any}[.]
- my name is {@PersonName.Any}[.]

## Logout
- forget me
- log out

## SelectItem
- choose last
- choose the {@DirectionalReference=bottom left}
- choose {@DirectionalReference=top right}
- i like {@DirectionalReference=left} one

## SelectNone
- none

@ ml DirectionalReference
@ prebuilt personName
```

Use the drop-down menus to select the answer choice that completes each statement based on the information presented in the graphic.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

`SelectItem` is **[answer choice]**.

- a domain
- an entity
- an intent
- an utterance

`Choose {@DirectionalReference=top right}` is **[answer choice]**.

- a domain
- an entity
- an intent
- an utterance

Correct Answer:

Answer Area

`SelectItem` is **[answer choice]**.

- a domain
- an entity
- an intent
- an utterance

`Choose {@DirectionalReference=top right}` is **[answer choice]**.

- a domain
- an entity
- an intent
- an utterance

Reference:
https://github.com/solliancenet/tech-immersion-data-ai/blob/master/ai-exp1/README.md

Question #13 *Topic 5*

HOTSPOT -

You are designing a conversation flow to be used in a chatbot.

You need to test the conversation flow by using the Microsoft Bot Framework Emulator.

How should you complete the .chat file? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
user=User1
bot=watchbot
user: I want a new watch.
bot: [ ▼ ][Delay=3000]
```

Attachment / ConversationUpdate / Typing

```
bot: I can help you with that! Let me see what I can find.
bot: Here's what I found.
bot:
[AttachmentLayout= ▼ ]
```

adaptivecard / carousel / thumbnail

```
[Attachment=https://contoso.blob.core.windows.net/watch01.jpg]
[Attachment=https://contoso.blob.core.windows.net/watch02.jpg]
user: I like the first one.
bot: Sure, pulling up more information.
bot: [Attachment=cards\watchProfileCard.json ▼ ]
user: That's nice! Thank you.
bot: Sure, you are most welcome!
```

adaptivecard / carousel / list

Correct Answer:

Answer Area

```
user=User1
bot=watchbot
user: I want a new watch.
bot: [ ▼ ][Delay=3000]
```

Attachment / ConversationUpdate / Typing

```
bot: I can help you with that! Let me see what I can find.
bot: Here's what I found.
bot:
[AttachmentLayout= ▼ ]
```

adaptivecard / carousel / thumbnail

```
[Attachment=https://contoso.blob.core.windows.net/watch01.jpg]
[Attachment=https://contoso.blob.core.windows.net/watch02.jpg]
user: I like the first one.
bot: Sure, pulling up more information.
bot: [Attachment=cards\watchProfileCard.json ▼ ]
user: That's nice! Thank you.
bot: Sure, you are most welcome!
```

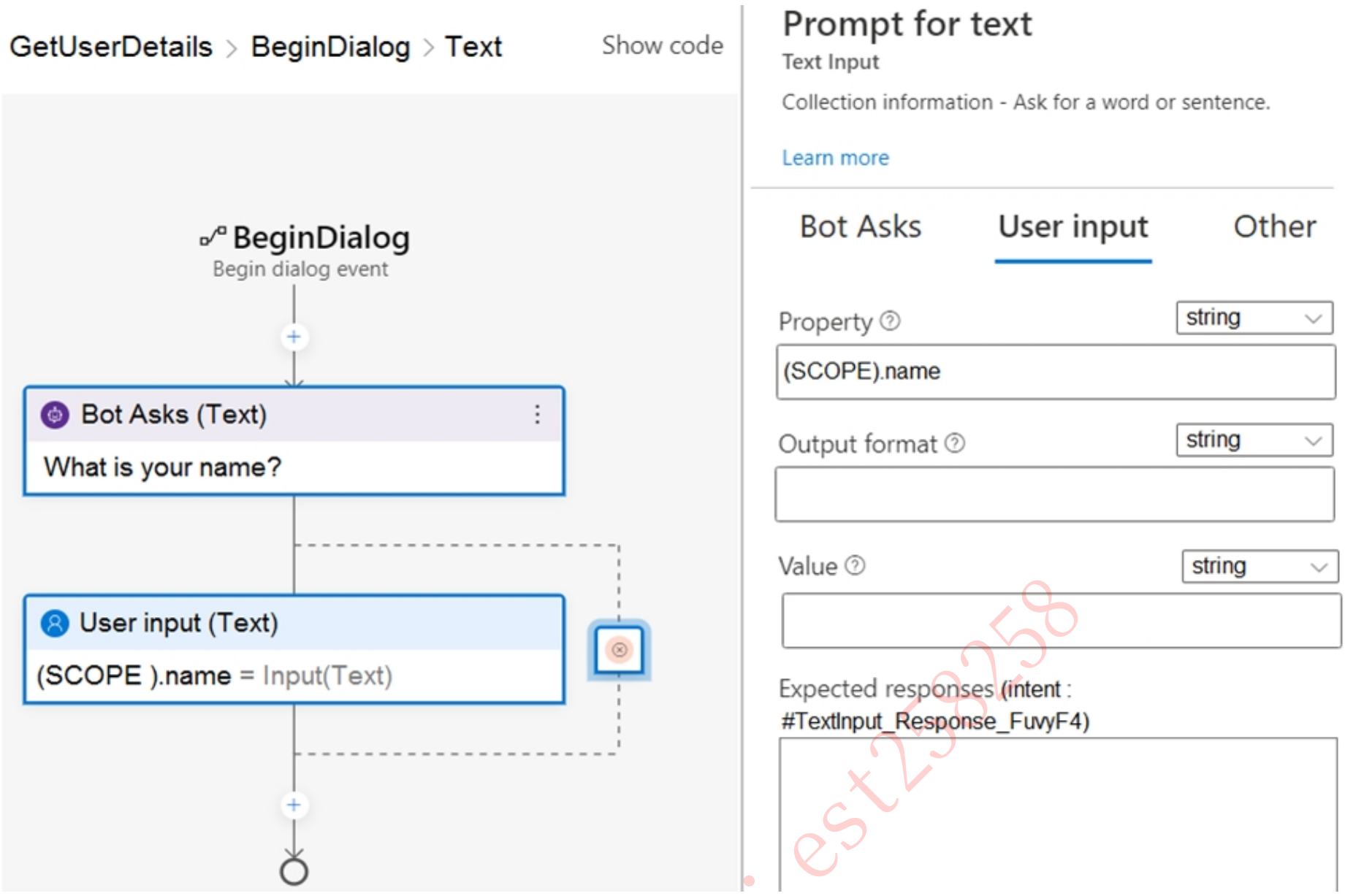
adaptivecard / carousel / list

Reference:

https://docs.microsoft.com/en-us/azure/bot-service/bot-builder-howto-add-media-attachments?view=azure-bot-service-4.0&tabs=csharp

Question #14 *Topic 5*

You are building a chatbot by using the Microsoft Bot Framework Composer as shown in the exhibit. (Click the Exhibit tab.)



The chatbot contains a dialog named GetUserDetails. GetUserDetails contains a TextInput control that prompts users for their name.

The user input will be stored in a property named name.

You need to ensure that you can dispose of the property when the last active dialog ends.

Which scope should you assign to name?

A. dialog **Most Voted**

B. user

C. turn

D. conversation

Correct Answer: *A*

Community vote distribution

A (100%)