A. Yes
B. No

Correct Answer: B
Explanation:
Use a separate Azure Event Grid topics and subscriptions for sign-in and sign-out events.

Scenario: Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

QUESTION 14
You need to ensure that the solution can meet the scaling requirements for Policy Service. Which Azure Application Insights data model should you use?

A. an Application Insights metric
B. an Application Insights dependency
C. an Application Insights trace
D. an Application Insights even

Correct Answer: A
Explanation:
Application Insights provides three additional data types for custom telemetry:

Trace - used either directly, or through an adapter to implement diagnostics logging using an instrumentation framework that is familiar to you, such as Log4Net or System.Diagnostics.

Event - typically used to capture user interaction with your service, to analyze usage patterns.

Metric - used to report periodic scalar measurements.

Scenario:
Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

References:
https://docs.microsoft.com/en-us/azure/azure-monitor/app/data-model

QUESTION 15
DRAG DROP
You need to add code at line EG15 in EventGridController.cs to ensure that the Log policy applies to all services.

How should you complete the code? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Code segments

topic
status
eventType
Succeeded
operationName
resourceProvider

Answer Area

```
if (
    @event[ "data" ] [ "_____"].ToString() = ="_____"
    &&
    @event[ "data" ] [ "_____"].ToString() = = "Microsoft.Web/sites/write"
)
```

Correct Answer:

Code segments

topic
status
eventType
Succeeded
operationName
resourceProvider

Answer Area

```
if (
    @event[ "data" ] [ "status"].ToString() = =" Succeeded "
    &&
    @event[ "data" ] [ "operationName"].ToString() = = "Microsoft.Web/sites/write"
)
```

Topic 4, Chatbot

Background
Best for You Organics Company is a global restaurant franchise that has multiple locations. The company wants to enhance user experiences and vendor integrations. The company plans to implement automated mobile ordering and delivery services.

Best For You Organics hosts an Azure web app at the URL https://www.bestforyouorganics.com.

Users can use the web app to browse restaurant locations, menu items, nutritional, information, and company information. The company developed and deployed a cross-platform mobile app.

Requirements
You must develop a chartbot by using the Bot Builder SDK and Language Understanding Intelligence Service (LUIS). The chatbot must allow users to order food for pickup or delivery.

The chatbot must meet the following requirements:

▪ Ensure that chatbot endpoint can be accessed only by the Bot Framework connector.
▪ Use natural language processing and speech recognition so that users can interact with the chatbot by using text and voice. Processing must be server-based.
▪ Alert users about promotions at local restaurants.
▪ Enable users to place an order for delivery or pickup by using their voice.
▪ Greet the user upon sigh-in by displaying a graphical interface that contains action buttons.
▪ The chatbot greeting interface must match the formatting of the following example:

Welcome to the Restaurant!

John Doe
Sun, Aug 26, 2018

Welcome to Best For You Organics Company! How can we help you today?

Specials: Chicken Marsala

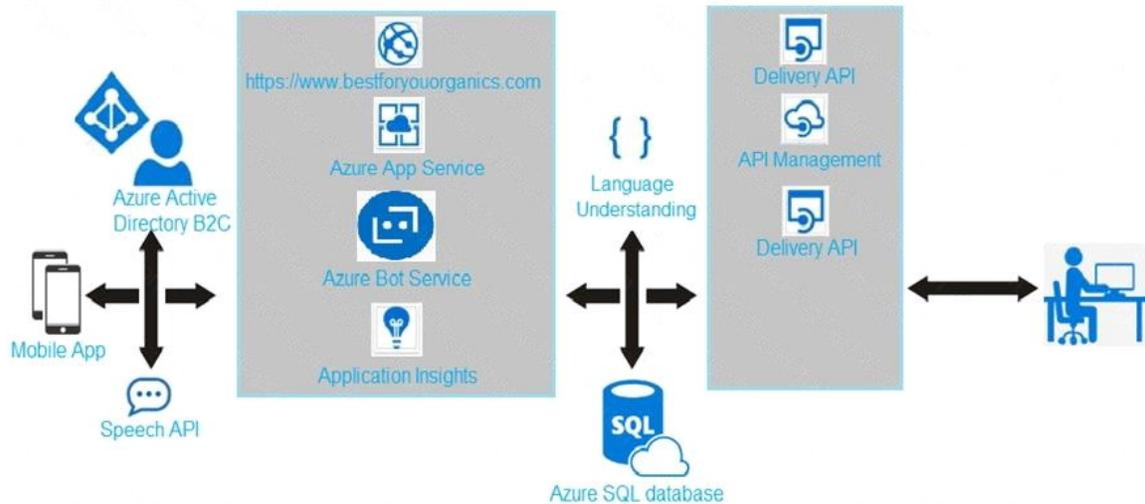**Order Pickup**   **Order Delivery**

Vendor API

Vendors receive and provide updates for the restaurant inventory and delivery services by using Azure API Management hosted APIs. Each vendor uses their own subscription to access each of the APIs.

APIs must meet the following conditions:

- API usage must not exceed 5,000 calls and 50,000 kilobytes of bandwidth per hour per vendor.
- If a vendor is nearing the number of calls or bandwidth limit, the API must trigger email notifications to the vendor.
- APIs must prevent API usage spikes on a per-subscription basis by limiting the call rate to 100 calls per minute.
- The Inventory API must be written by using ASP.NET Core and Node.js.
- The API must be updated to provide an interface to Azure SQL Database. Database objects must be managed by using code.
- The Delivery API must be protected by using the OAuth 2.0 protocol with Azure Active Directory (Azure AD) when called from the Azure web app. You register the Delivery API and web app in Azure AD. You enable OAuth 2.0 in the web app.
- The delivery API must update the Products table, the Vendor transactions table, and the Billing table in a single transaction.

The Best For You Organics Company architecture team has created the following diagram depicting the expected deployments into Azure:

Delivery API

The Delivery API intermittently throws the following exception:

```
"System.Data.Entity.Core.EntityCommandExecutionException: An error occurred while executing the
command definition. See the inner exception for details. --->System.Data.SqlClient.SqlException:
A transport-level error has occurred when receiving results from the server. (provider: Session
Provider, error: 19 - Physical connection is not usable)"
```

Chatbot greeting

The chatbot's greeting does not show the user's name. You need to debug the chatbot locally.

Language processing

Users report that the bot fails to understand when a customer attempts to order dishes that use Italian names.

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

Startup.cs

```
SU01   namespace DeliveryApi
SU02   {
SU03    public class Startup
SU04    {
SU05     public Startup (Iconfiguration configuration)
SU06     {
SU07      Configuration = configuration ;
SU08     }
SU09     public Iconfiguration Configuration {get ;}
SU10     public void ConfigureServices(IserviceCollection services)
SU11   {
SU12    services.AddDbContext<RestaurantsContext> (opt =>
SU13      opt.UseSqlServer (Configuration.GetSection ("ConnectionStrings") ["RestaurantsDatabase"],
SU14       sqlServerOptionsAction: sqlOptions =>
SU15       {
SU16        . . .
SU17   }))) ;
SU18   services.AddMvc()
SU19   .SetCompatibilityVersion(CompatibilityVersion.Version_2_1) ;
SU20   }
SU21   public void Configure (IapplicationBuilder app)
SU22    {
SU23     app.UseMvc() ;
SU24    }
SU25  }
SU26 }
```

QUESTION 1
Note: This question is part of a series of questions that present the same scenario.

Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You need to meet the vendor notification requirement.

Solution: Update the Delivery API to send emails by using a cloud -based email service.

Does the solution meet the goal?

A. Yes
B. No

Correct Answer: B

QUESTION 2
Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution.

Determine whether the solution meets the stated goals.

You need to meet the vendor notification requirement.

Solution: Create and apply a custom outbound Azure API Management policy.

Does the solution meet the goal?