5. Connect to and consume Azure services and third-party services
└ **5.3 Develop message-based solutions**
└ **5.3.1 Implement solutions that use Azure Service Bus**

1. What is Azure Service Bus and what are its use cases?
2. What is the difference between Service Bus, Event Hubs, and Event Grid?
3. What messaging models does Service Bus support?
4. What is the difference between standard and premium tiers in Service Bus?
5. What are message sessions and when are they needed?
6. What is dead-lettering and how is it configured?
7. How do you send messages using Azure SDK or CLI?
8. How do you receive messages from a queue?
9. What is peek-lock vs receive-and-delete mode?
10. How do you implement message deferral and why?
11. What is auto-forwarding in Service Bus?
12. How do you configure filters in topic subscriptions?
13. How do you authenticate and authorize access to Service Bus?
14. What are delivery and retry behaviors in Service Bus?
15. How do Service Bus and Azure Functions integrate?

---

1. What is Azure Service Bus and what are its use cases?
A fully managed message broker for enterprise apps.
Use cases: decoupled microservices, order processing, transactions, retries, delayed delivery.

---

2. What is the difference between Service Bus, Event Hubs, and Event Grid?
- **Service Bus**: Reliable messaging, ordering, sessions, dead-lettering.
- **Event Hubs**: High-throughput telemetry streaming.
- **Event Grid**: Lightweight, push-based eventing.

---

3. What messaging models does Service Bus support?
- **Queues**: Point-to-point (1 sender → 1 receiver).
- **Topics/Subscriptions**: Publish-subscribe (1 sender → multiple filtered subscribers).

---

4. What is the difference between standard and premium tiers in Service Bus?
- **Standard**: Basic features, shared resources, limited performance.
- **Premium**: Dedicated compute, faster, supports VNET, encryption, higher scale.

---

5. What are message sessions and when are they needed?
Sessions group related messages for ordered processing.
Required when strict message ordering per entity is needed (e.g., per user/cart).

---

6. What is dead-lettering and how is it configured?
Dead-letter queue (DLQ) stores undeliverable messages (e.g., max delivery attempts reached).
Enable via `EnableDeadLetteringOnMessageExpiration` or use `Abandon`/`DeadLetter` in SDK.

7. How do you send messages using Azure SDK or CLI?

.NET SDK:

```
await sender.SendMessageAsync(new ServiceBusMessage("Hello"));
```

CLI:

```
az servicebus message send --queue-name myqueue --namespace-name myns --resource-group myrg --body "msg"
```

---

8. How do you receive messages from a queue?

.NET SDK:

```
var msg = await receiver.ReceiveMessageAsync();
await receiver.CompleteMessageAsync(msg);
```

---

9. What is peek-lock vs receive-and-delete mode?

- **Peek-lock** (default): Two-phase – lock then explicitly complete.
- **Receive-and-delete**: One-shot; message is deleted on receive.

---

10. How do you implement message deferral and why?

Deferred messages are postponed for later retrieval by sequence number.
Useful when processing must be delayed:

```
var deferred = await receiver.ReceiveDeferredMessageAsync(sequenceNumber);
```

---

11. What is auto-forwarding in Service Bus?

Automatically forwards messages from one queue/topic to another.
Used for message routing, chaining processing steps.

---

12. How do you configure filters in topic subscriptions?

Use SQL-based filters:

```
az servicebus topic subscription rule create \
  --name highPriority \
  --subscription-name mysub \
  --topic-name mytopic \
  --filter-sql-expression "priority = 'high'"
```

---

13. How do you authenticate and authorize access to Service Bus?

- **Shared Access Signature (SAS)**: via policy keys and connection string.
- **Azure AD**: via RBAC roles like `Azure Service Bus Data Sender`.

---

14. What are delivery and retry behaviors in Service Bus?

- Default: 10 delivery attempts, exponential backoff.
- Messages that fail retry go to DLQ if enabled.
- Customize via `MaxDeliveryCount` and `LockDuration`.

---

15. How do Service Bus and Azure Functions integrate?

Use Service Bus trigger:

```
[ServiceBusTrigger("myqueue", Connection = "ServiceBusConnection")] string msg
```