5. Connect to and consume Azure services and third-party services
└ 5.3 Develop message-based solutions
└ 5.3.2 – Implement solutions that use Azure Queue Storage queues

1. What is Azure Queue Storage and what are its typical use cases?
2. How does Azure Queue Storage compare to Azure Service Bus queues?
3. What are the key components of an Azure Storage queue?
4. How are messages added to and retrieved from a queue?
5. What is the default behavior when retrieving messages from a queue?
6. What is message visibility timeout and how does it affect processing?
7. How do you handle message retries and poison messages in Azure Queue Storage?
8. How do you create and manage queues using Azure CLI or SDK?
9. How do you secure access to Azure Queue Storage (SAS tokens, RBAC, shared keys)?
10. What are the message size limits and encoding requirements?
11. How do you configure and use Base64 encoding with queue messages?
12. What are best practices for processing at scale with Queue Storage?
13. How do you integrate Azure Queue Storage with Azure Functions?
14. How do you monitor and troubleshoot Azure Queue Storage usage?
15. What are common limitations of Azure Queue Storage compared to other messaging services?

---

1. What is Azure Queue Storage and what are its typical use cases?
A simple, cloud-based message queueing service for decoupling components.
Use cases: task queues, background jobs, retry buffers.

---

2. How does Azure Queue Storage compare to Azure Service Bus queues?

- **Queue Storage**: Basic FIFO queues, lightweight, no ordering guarantees, no DLQ, no sessions.
- **Service Bus**: Enterprise-grade, supports DLQ, ordering, transactions, filtering.

---

3. What are the key components of an Azure Storage queue?

- **Queue**: A named message container.
- **Message**: Max 64 KB of Base64-encoded text.
- **Visibility timeout**: Temporary hide period after retrieval.

---

4. How are messages added to and retrieved from a queue?
Add: `PutMessageAsync()`
Get: `GetMessagesAsync()`
Delete: `DeleteMessageAsync()` after processing.

---

5. What is the default behavior when retrieving messages from a queue?
Messages are hidden for the visibility timeout duration.
If not deleted within that time, they become visible again for reprocessing.

---

6. What is message visibility timeout and how does it affect processing?
Defines how long a retrieved message stays hidden.
If not deleted in time, it is returned to the queue for another attempt.

7. How do you handle message retries and poison messages?

Track dequeue count using `DequeueCount`.

Manually delete or move messages to a custom "poison" queue if `DequeueCount` exceeds threshold.

---

8. How do you create and manage queues using Azure CLI or SDK?

CLI:

```
az storage queue create --name myqueue --account-name mystorage
```

.NET SDK:

```
await queueClient.CreateIfNotExistsAsync();
await queueClient.SendMessageAsync("my message");
```

---

9. How do you secure access to Azure Queue Storage?

- **Shared keys**: Full access via storage account key.
- **SAS tokens**: Scoped, time-limited access.
- **RBAC**: Role-based access via Azure AD (Blob/Queue Contributor roles).

---

10. What are the message size limits and encoding requirements?

- Max message size: **64 KB** (48 KB when Base64 encoded by default).
- Messages must be **text-based** (UTF-8 or Base64).

---

11. How do you configure and use Base64 encoding with queue messages?

By default, SDK encodes messages as Base64.

Can opt out using:

```
new QueueClient(..., new QueueClientOptions { MessageEncoding = QueueMessageEncoding.None })
```

---

12. What are best practices for processing at scale with Queue Storage?

- Use multiple consumer instances.
- Tune visibility timeout per job duration.
- Monitor queue length for autoscaling triggers.

---

13. How do you integrate Azure Queue Storage with Azure Functions?

Use the `QueueTrigger` binding:

```
[FunctionName("ProcessQueue")]
public void Run([QueueTrigger("myqueue", Connection = "StorageConnection")] string message)
```

---

14. How do you monitor and troubleshoot Azure Queue Storage usage?

- Enable diagnostics in Azure Monitor.
- Use metrics: `QueueLength`, `DequeueCount`, `Ingress`, `Egress`.
- View activity logs and errors in Log Analytics if enabled.

---

15. What are common limitations of Azure Queue Storage compared to other messaging services?

- No message ordering guarantee.
- No native dead-letter queues.
- No message filtering or pub/sub.
- Limited throughput and features compared to Service Bus.