# Microsoft Azure WebJobs SDK Quick Reference

## 1. Blob Operations

### 1.1. Simple Blob Trigger

Triggered when a new blob is created or updated in the container "output" with the same name. Binds the second argument to a blob in a container "output" with the same name.

```csharp
public static void BlobCopySimple(
    [BlobTrigger("{name}.{extension}", FileAccess.Write)] Stream output);
```

### 1.2. Advanced Blob Trigger

Triggered when a new blob is created or modified. The "name" and "extension" arguments are bound to parameters captured by the triggering blob path pattern. The "BlobTrigger" is bound to the triggering blob path. The "output" argument is bound to a write stream associated with a blob in "output" container with the same name.

```csharp
public static async Task BlobTriggerAdvanced(
    [BlobTrigger("{staging}.{name}.{extension}")] CloudBlockBlob blob,
    string name,
    string extension,
    string blobTrigger,
    [Blob("{output}.{name}.{extension}")], FileAccess.Write[]) Stream output);
```

### 1.3. Handle Poison Blob

Invoked when a blob triggering message ends up in the poison queue after five unsuccessful retries. The "log" parameter is bound to a text writer associated with a "log" output.

```csharp
public static void BlobCopySimple(
    [BlobTrigger("{name}.{extension}")] Stream output);
```

## 2. Queue Operations

### 2.1. Simple Queue Trigger

Triggered when a new message received from "orders" queue. Binds the second argument to a message in the container "output" queue.

```csharp
public static void QueueCopySimple(
    [QueueTrigger("orders")] CloudQueueMessage message,
    [Queue("output")] out string output);
```

### 2.2. Property Binding

Shows binding parameters to properties of a queue message. The "name" parameter is bound to the value of the "Name" property in the Order object. The "dequeueCount" parameter is bound to the property value of the actual CloudQueueMessage object.

```csharp
public static void QueueCopySimple(
    [QueueTrigger("orders")] Order order,
    string name,
    int dequeueCount,
    string queueTrigger,
    [Blob("{orders/ordersorderId}")]) out string orderData,
    TextReader log);
```

### 2.3. Handle Poison Message

Invoked when a message ends up in the poison queue. It is used to demonstrate the poison message messaging behavior. The trigger message is moved into a poison queue after a binding or a function fails 5 times.

```csharp
public static void PoisonQueueHandler(
    [QueueTrigger("orders-poison")] string poisonMessage);
```

## 3. Table Operations

### 3.1. Simple Table Update

Triggered on new message in the "orders" queue. Binds "entries" parameter to the "transactions" table.

```csharp
public static void TableUpdate(
    [Table("items")] ICollector<OrderItem> items);
```

### 3.2. Manual Trigger

Not triggered automatically. Binds "logEntries" to the "log" table.

```csharp
public static void TableUpdate(
    [Table("items")] ICollector<OrderItem> items);
```

## 4. ServiceBus Operations

### 4.1. Simple ServiceBus Queue Trigger

Triggered when a new message received from "incoming" queue. Binds the "message" argument to outgoing message in "outgoing" queue.

```csharp
public static void ServiceBusMessageCopySimple(
    [ServiceBusTrigger("incoming")] string message,
    [ServiceBus("outgoing")] out string message);
```

### 4.2. Simple ServiceBus Topic Listener

Topic subscription listener.

```csharp
public static void ServiceBusMessageCopySimple(
    [ServiceBusTrigger("incoming")] string message,
    [ServiceBus("outgoing")] out string message);
```

## 5. Misc Operations

### 5.1. Runtime Binder

Shows binding to Binder for runtime binding.

```csharp
public static void ProcessRequestWithRuntimeBinder(
    [QueueTrigger("requests")] Request request,
    Binder binder);
```

### 5.2. Cancellation Token

Shows binding to a cancellation token. Triggered by a queue message received from a queue with a name resolved by NameResolver.

```csharp
public static void ShutdownMonitor(
    [QueueTrigger("shutdownQueueName")] string message,
    CancellationToken cancelToken);
```

As de-serialized by ICloudBlobStreamBinder<T>. See notes on stream read binding behavior.

The default value is null. If none specified then explicit FileAccess attribute property is required. See "Bound by <CloudBlobStreamBinder<T>>." See notes on stream write binding behavior.

For more information about the WebJobs feature of Microsoft Azure Web Sites, see http://go.microsoft.com/fwlink/?Linkid=390226.