What is Interrupt Testing?
Interrupt testing deals with how an application reacts to interrupt and resume to it's previous state.

Type of Interruptions in Mobile Application

- Battery low
- Battery full- when charging
- Incoming phone call
- Incoming SMS
- Incoming Alert from another mobile application
- Plugged in for charging
- Plugged out from charging
- Device shut off
- Application Update reminders
- Alarm
- Network connection loss
- Network connection restoration

This list is not exhaustive but includes the most common scenarios.

Resolution in case of Interrupt:

The expected behaviour in case of these interruptions is one of the following:

1.Run in background: The interruption takes over while the application takes a back seat. It gains control after the interruption ends. For example, A phone call/Facetime that you attend while you are reading a digital book on iBooks(or similar application).
When the user answers a phone, iBooks waits until it is done and then resumes when the call ends.

2.Show alert. Alert disappears, and you work as usual: 'SMS received'- messages appear in the header. The user doesn't bother about it and continue working with the application as normal. Other mobile app alerts, such as a new friend request on Facebook or WhatsApp message, also fall into this category. But if the user decides to read the message, the behaviour described in Point 1 is followed. If ignored, the application's state is unchanged.

3.Call to Action: Alarms have to be turned off or snoozed before you continue working. Same thing with App update messages. You either have to Cancel or Accept thechanges before you proceed. Another example is that of the low battery alert- You can choose to continue as usual or go into a low power mode (if the device allows it.)

4.No impact: An example is: if a network connection becomes available and your device connects to it. Also, when you plug your device in for charging, no alert or call to action step is necessary. It will probably do its job while you continue using your application.

Multi-touch or Single touch:

If your app is supporting the multi-touch feature like pinch to zoom or pinch to shrink etc., then you need to thoroughly test this feature and create a lot of test cases for this for all the applicable screens.