

Java Event Handling Interview Questions and Answers Pdf

Question: 1

What is an adapter class?

An adapter class provides an empty implementation of all methods in an event listener interface.

Adapter classes are useful when you want to receive and process only some of the events that are handled by a particular event listener interface.

You can define a new class to act listener by extending one of the adapter classes and implementing only those events in which you are interested.

For example, the MouseAdapter class has two methods, mouseDragged() and mouseMoved().

Question: 2

What interface is extended by AWT event listener?

All AWT event listeners extend the java.util.EventListener interface.

Question: 3

What is an event and what are the models available for event handling?

An event is an event object that describes a state of change in a source.

In other words, event occurs when an action is generated, like pressing button, clicking mouse, selecting a list, etc.

There are two types of models for handling events and they are

event inheritance model and

event delegation model.

Question: 4

What is the purpose of the enableEvents() method?

The enableEvents() method is used to enable an event for a particular object.

Normally, an event is enabled when a listener is added to an object for a particular event.

The enableEvents() method is used by objects that handle events by overriding their event dispatch methods.

Question: 5

What are the advantages of the model over the event inheritance model?

The event delegation model has two advantages over the event inheritance model. They are

It enables event handling by objects other than the ones that generate the events.

This allows a clean separation between a components design and its use.

It performs much better in applications where many events are generated.

This performance improvement is due to the fact that the event delegation model does not have to be repeatedly process unhandled events as is the case of the event inheritance.