

| | |
|----------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>-(NSResponder *)nextResponder</code> | Returns the receiver's next responder. |
| <code>-(void)setNextResponder:(NSResponder *)aResponder</code> | Makes aResponder the receiver's next responder. |
| <code>-(BOOL)acceptsFirstResponder</code> | Subclasses override to accept or reject first responder status. NSResponder's implementation simply returns NO. |
| <code>-(void)becomeFirstResponder</code> | Notifies the receiver that it's the first responder. |
| <code>-(void)resignFirstResponder</code> | Notifies the receiver that it's not the first responder. |
| <code>-(void)performKeyEquivalent:(NSEvent *)theEvent</code> | Subclasses override to respond to keyboard input. NSResponder's implementation simply returns NO to indicate theEvent isn't handled. |
| <code>-(void)tryToPerform:(SEL)anAction with:(id)anObject</code> | Aids in dispatching action messages. Returns YES if an responder in the responder chain can perform the anAction method, which takes the single argument anObject. |
| <code>-(void)flagsChanged:(NSEvent *)theEvent</code> | Subclasses override to handle flags-changed events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-(void)helpRequested:(NSEvent *)theEvent</code> | Causes the Help panel to display the help attached to the receiver. If there's no attached help, passes the message to the receiver's next responder. |
| <code>-(void)keyDown:(NSEvent *)theEvent</code> | Subclasses override to handle key-down events. NSResponder's implementation passes the message to the receiver's next responder. If the first responder changes, this method posts the notification <code>NSNotification</code> with the current object and, in the notification's dictionary, the key <code>NSStringMovement</code> to the default notification center. |
| <code>-(void)keyUp:(NSEvent *)theEvent</code> | Subclasses override to handle key-up events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-(void)mouseDown:(NSEvent *)theEvent</code> | Subclasses override to handle mouse-down events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-(void)mouseDragged:(NSEvent *)theEvent</code> | Subclasses override to handle mouse-dragged events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-(void)mouseEntered:(NSEvent *)theEvent</code> | Subclasses override to handle mouse-entered events. NSResponder's implementation passes the message to the receiver's next responder. |

| | |
|------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | generates a beep. |
| <code>-rightMouseDown:(NSEvent *)theEvent</code> | Subclasses override to handle right mouse-down events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-rightMouseDownDragged:(NSEvent *)theEvent</code> | Subclasses override to handle right mouse-dragged events. NSResponder's implementation passes the message to the receiver's next responder. |
| <code>-rightMouseUp:(NSEvent *)theEvent</code> | Subclasses override to handle right mouse-up events. NSResponder's implementation passes the message to the receiver's next responder. |
| | |
| <code>-validRequestorForSendType:(NSString *)typeSent returnType:(NSString *)typeReturned</code> | Subclasses override to determine which Services menu items are available at a given time. Returning self enables services that can receive the given pasteboard types and can return typeReturned pasteboard types. Returning nil disables them. NSResponder's implementation passes the message to the receiver's next responder. |