

Dialogs in Flutter

In Flutter, dialogs are used to display a modal window that requires user interaction. Flutter provides several built-in dialogs like `AlertDialog`, `SimpleDialog`, and `Dialog`, as well as the ability to create custom dialogs.

1. AlertDialog

An `AlertDialog` is a simple dialog with an optional title, content, and actions. It's typically used to alert the user or confirm an action.

Example:

```
showDialog(  
  context: context,  
  builder: (BuildContext context) {  
    return AlertDialog(  
      title: Text('Alert'),  
      content: Text('This is an alert dialog.'),  
      actions: <Widget>[  
        TextButton(  
          child: Text('Cancel'),  
          onPressed: () {  
            Navigator.of(context).pop(); // Dismiss the dialog  
          },  
        ),  
        TextButton(  
          child: Text('OK'),  
          onPressed: () {
```

```

        // Do something on OK pressed

        Navigator.of(context).pop(); // Dismiss the dialog

    },

),

],

);

},

);

```

2. SimpleDialog

A SimpleDialog is a less customizable dialog that allows for a list of options, each represented by a SimpleDialogOption. It is typically used for presenting a list of choices.

Example:

```

showDialog(

  context: context,

  builder: (BuildContext context) {

    return SimpleDialog(

      title: Text('Select an option'),

      children: <Widget>[

        SimpleDialogOption(

          onPressed: () {

            // Do something when the option is selected

            Navigator.of(context).pop();

          },

          child: Text('Option 1'),

        ),

```

```

SimpleDialogOption(

  onPressed: () {

    // Do something when the option is selected

    Navigator.of(context).pop();

  },

  child: Text('Option 2'),

),

],

);

},

);

```

3. Custom Dialog

If you need more customization, you can create your own dialog by using the Dialog widget.

Example:

```

showDialog(

  context: context,

  builder: (BuildContext context) {

    return Dialog(

      shape: RoundedRectangleBorder(

        borderRadius: BorderRadius.circular(20.0),

      ),

      child: Container(

        height: 200,

        child: Padding(

          padding: const EdgeInsets.all(12.0),

```

```

child: Column(

  mainAxisAlignment: MainAxisAlignment.center,

  crossAxisAlignment: CrossAxisAlignment.start,

  children: [

    Text('Custom Dialog'),

    SizedBox(height: 24),

    Row(

      mainAxisAlignment: MainAxisAlignment.end,

      children: [

        TextButton(

          onPressed: () {

            Navigator.of(context).pop(); // Dismiss the dialog

          },

          child: Text('Close'),

        ),

      ],

    ),

  ],

),

),

),

);

},

);

```

4. Bottom Sheet Dialog

For a dialog that slides up from the bottom of the screen, you can use a `BottomSheet`.

Example:

```
showModalBottomSheet(  
  
  context: context,  
  
  builder: (BuildContext context) {  
  
    return Container(  
  
      height: 200,  
  
      child: Center(  
  
        child: Text('This is a bottom sheet'),  
  
      ),  
  
    );  
  
  },  
  
);
```

Handling Dialog Results

You can also handle the result returned by the dialog using `Navigator.pop`.

Example:

```
Future<void> _showMyDialog() async {  
  
  final result = await showDialog(  
  
    context: context,  
  
    builder: (BuildContext context) {  
  
      return AlertDialog(  
  
        title: Text('Confirm'),  
  
        content: Text('Are you sure?'),  
  
        actions: <Widget>[  
  
          TextButton(  

```

```

    child: Text('No'),

    onPressed: () {

      Navigator.of(context).pop(false); // Return false

    },

  ),

  TextButton(

    child: Text('Yes'),

    onPressed: () {

      Navigator.of(context).pop(true); // Return true

    },

  ),

],

);

},

);

```

```

if (result == true) {

  // Do something if Yes is selected

} else {

  // Do something if No is selected

}

}

```

These are the basics of using dialogs in Flutter. You can customize them further based on your app's requirements.