

```

// stateful.dart

import 'package:flutter/material.dart';

import '../reduceable.dart';

typedef ReduceableWidgetBuilder<S> = Widget Function(
  Reduceable<S> value,
  Widget child,
);

class StateProvider<S> extends StatefulWidget {
  const StateProvider({
    super.key,
    required this.state,
    required this.child,
    required this.builder,
  });

  final S state;
  final Widget child;
  final ReduceableWidgetBuilder<S> builder;

  @override
  State<StateProvider> createState() => _StateProviderState<S>(state);
}

class _StateProviderState<S> extends State<StateProvider<S>> {
  _StateProviderState(S state) : _state = state;

  S _state;

  void reduce(Reducer<S> reducer) =>
    setState(() => _state = reducer(_state));

  @override
  Widget build(BuildContext context) => widget.builder(
    Reduceable(_state, reduce),
    widget.child,
  );
}

// </br>

```

```

class InheritedValueWidget<V> extends InheritedWidget {
  const InheritedValueWidget({
    super.key,
    required super.child,
    required this.value,
  });

  final V value;

  static V of<V>(BuildContext context) =>
    _widgetOf<InheritedValueWidget<V>>(context).value;

  static W _widgetOf<W extends InheritedValueWidget>(
    BuildContext context) =>
    context.dependOnInheritedWidgetOfExactType<W>()!;

  @override
  bool updateShouldNotify(InheritedValueWidget oldWidget) =>
    value != oldWidget.value;
}

```