

```
// bloc_binder.dart

import 'package:flutter/widgets.dart';
import 'package:flutter_bloc/flutter_bloc.dart';

import '../domain.dart';
import '../builder.dart';
import '../reduceable.dart';
import 'bloc.dart';

class MyAppStateBinder extends StatelessWidget {
  const MyAppStateBinder({Key? key, required this.child})
    : super(key: key);

  final MyAppState state = const MyAppState(
    title: 'Flutter Demo Home Page',
    counter: 0,
  );
  final Widget child;

  @override
  Widget build(BuildContext context) => BlocProvider(
    create: (_) => MyAppStateBloc(),
    child: child,
  );
}

class MyHomePageBinder extends StatelessWidget {
  const MyHomePageBinder({super.key});

  @override
  Widget build(context) =>
    BlocSelector<MyAppStateBloc, MyAppState, MyHomePageProps>(
      selector: (state) => MyHomePageProps.reduceable(
        context.appStateBloc.reduceable,
      ),
      builder: (context, props) => MyHomePageBuilder(
        props: props,
      ),
    );
}

// </br>
```

```

class MyCounterWidgetBinder extends StatelessWidget {
  const MyCounterWidgetBinder({super.key});

  @override
  Widget build(context) =>
    BlocSelector<MyAppStateBloc, MyAppState, MyCounterWidgetProps>(
      selector: (state) => MyCounterWidgetProps.reduceable(
        context.appStateBloc.reduceable,
      ),
      builder: (context, props) => MyCounterWidgetBuilder(
        props: props,
      ),
    );
}

extension _MyAppStateBlocOnBuildContext on BuildContext {
  MyAppStateBloc get appStateBloc =>
    BlocProvider.of<MyAppStateBloc>(this);
}

extension _ReduceableOnMyAppStateBloc on MyAppStateBloc {
  Reduceable<MyAppState> get reduceable =>
    Reduceable(state, add);
}

```