**State Management Module 1**

**WRAP UP**

First we use the state management system in our application to clear our main.dart file and also organize our code using provders pattern which uses listener to build specific area of our application. So we use provider package in main.dart file by providing objects of the widget like this

return MultiProvider(providers: [  
 ChangeNotifierProvider(//this is a provider  
 create:(ctx)=> Products(),  
 ),  
 ChangeNotifierProvider(  
 create: (ctx)=> Cart(),  
 ),  
 ChangeNotifierProvider(create: (ctx)=> Orders(),  
 ),  
],

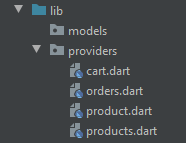
Also we use consumer in the product overview screen

Consumer<Cart>(builder: (\_, cart, ch) => Badge(  
 child: ch ,  
 value: cart.itemCount.toString(),  
),  
child: IconButton(  
 icon: Icon(  
 Icons.*shopping\_cart*,  
 ),  
 onPressed:(){  
 Navigator.*of*(context).pushNamed(CartScreen.*routeName*);  
 } ,  
 ),  
),

The difference between consumer and the providers.of . provider.of you rerun the entire build app even if you in a stateless widget. With the consumer it is the bit different you also rerun when ever the data changes so consumer is nice when you only run the specific part of the widget tree instead of the entire widget tree.

**Recalibrate structure :**

First we create providers folder because we using providers patterns. And make 4 dart file also download providers package ad set in pubsey.yaml



**Product.dart:**

We use change notifier so they use in entire widget.

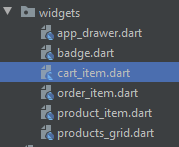
import 'package:flutter/foundation.dart';  
  
class Product with ChangeNotifier {  
 final String id;  
 final String title;  
 final String description;  
 final double price;  
 final String imageUrl;  
 bool isFavorite;  
  
 Product(  
 {@required this.id,  
 @required this.title,  
 @required this.description,  
 @required this.price,  
 @required this.imageUrl,  
 this.isFavorite = false});  
  
 void toggleFavoriteStatus(){  
 isFavorite = !isFavorite;  
 notifyListeners();  
 }  
}

**Products.dart:**

In this file we make list of array of product and make a method of get product using spread operator and also make a method find by id when we click any product so they show the name image description of product using id.

import 'package:flutter/cupertino.dart';  
 import 'product.dart';  
  
  
class Products with ChangeNotifier {  
 List<Product> \_items = [  
 Product(  
 id: 'p1',  
 title: 'Red Shirt',  
 description: 'A red shirt - it is pretty red!',  
 price: 29.99,  
 imageUrl:  
 'https://cdn.pixabay.com/photo/2016/10/02/22/17/red-t-shirt-1710578\_1280.jpg',  
 ),  
 Product(  
 id: 'p2',  
 title: 'Trousers',  
 description: 'A nice pair of trousers.',  
 price: 59.99,  
 imageUrl:  
 'https://upload.wikimedia.org/wikipedia/commons/thumb/e/e8/Trousers%2C\_dress\_%28AM\_1960.022-8%29.jpg/512px-Trousers%2C\_dress\_%28AM\_1960.022-8%29.jpg',  
 ),  
 Product(  
 id: 'p3',  
 title: 'Yellow Scarf',  
 description: 'Warm and cozy - exactly what you need for the winter.',  
 price: 19.99,  
 imageUrl:  
 'https://live.staticflickr.com/4043/4438260868\_cc79b3369d\_z.jpg',  
 ),  
 Product(  
 id: 'p4',  
 title: 'A Pan',  
 description: 'Prepare any meal you want.',  
 price: 49.99,  
 imageUrl:  
 'https://upload.wikimedia.org/wikipedia/commons/thumb/1/14/Cast-Iron-Pan.jpg/1024px-Cast-Iron-Pan.jpg',  
 ),  
 Product(  
 id: 'p5',  
 title: 'watches',  
 description: 'Best rolex watch',  
 price: 12.99,  
 imageUrl:  
 'https://images-na.ssl-images-amazon.com/images/I/71ww5fmHMpL.\_AC\_UX522\_.jpg'  
 ),  
  
  
 ];  
  
  
  
 List<Product> get items {  
  
 return [...\_items];  
 }  
  
 List<Product> get favoriteItems {  
 return \_items.where((prodItem) => prodItem.isFavorite).toList();  
 }  
 Product findById(String id){  
 return \_items.firstWhere((prod) => prod.id == id);  
 }  
  
  
  
 void addProduct(){  
 // \_items.add(value);  
 notifyListeners();  
 }  
  
  
}

Now add a widget folder and add a dart files.



**Product\_grid.dart:**

In this dart file we create the gridview builder also we provide the providers in this file and set the product length also we set the child : ProductItem() so this is another widget where we set header footer and icon of grid card .

import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
  
import '../providers/products.dart';  
import './product\_item.dart';  
  
class ProductsGrid extends StatelessWidget {  
 final bool showFavs;  
  
 ProductsGrid(this.showFavs);  
 @override  
 Widget build(BuildContext context) {  
 final productsData = Provider.*of*<Products>(context);  
 final products = showFavs ? productsData.favoriteItems : productsData.items;  
 return GridView.builder(  
 // padding: const EdgeInsets.all(10.0),  
 padding: const EdgeInsets.fromLTRB(10.0, 10.0, 10.0, 10.0),  
  
  
 itemCount: products.length,  
 itemBuilder: (ctx, i) => ChangeNotifierProvider.value(  
 value: products[i],  
 child: ProductItem(),  
 ),  
  
 gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(  
 crossAxisCount: 1,  
 childAspectRatio: 3 / 2,  
 crossAxisSpacing: 90,  
 mainAxisSpacing: 90,  
 ),  
 );  
 }  
}

**product\_item.dart:**

Where we set the header and the footer of the gridlist of the

import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
import 'package:shop\_project/providers/cart.dart';  
import 'package:shop\_project/providers/product.dart';  
import '../screens/product\_detail\_screen.dart';  
import '../providers/cart.dart';  
import '../providers/products.dart';  
  
class ProductItem extends StatelessWidget {  
  
  
 @override  
 Widget build(BuildContext context) {  
 final product = Provider.*of*<Product>(context);  
 final cart = Provider.*of*<Cart>(context, listen: false);  
  
  
 return ClipRRect(  
  
 //ClipRect help clipping rectangle round corner  
 borderRadius: BorderRadius.circular(10),  
  
 child: GridTile(  
  
 //Grid tile is use as header card for display image  
  
 child: GestureDetector(  
 onTap: () {  
 Navigator.*of*(context).pushNamed(  
 ProductDetailScreen.*routeName*,  
 arguments: product.id ,  
 );  
 },  
 child: Image.network(  
 product.imageUrl,  
 fit: BoxFit.cover,  
 ),  
 ),  
  
 footer: GridTileBar(  
 //gridtile is a textbar of product name  
 backgroundColor: Colors.*black54*,  
 leading: IconButton(  
 // leading make the icon left  
 icon: Icon(product.isFavorite ? Icons.*favorite*: Icons.*favorite\_border*,),  
 color: Theme.*of*(context).accentColor,  
  
 onPressed: () {  
 product.toggleFavoriteStatus();  
 },  
 ),  
  
 trailing: IconButton(  
 //trailing make the icon right  
 icon: Icon(Icons.*shopping\_cart*),  
 onPressed: () {  
 cart.addItem(product.id, product.price, product.title);  
 },  
 color: Theme.*of*(context).accentColor,  
 ),  
 title: Text(  
 product.title,  
 textAlign: TextAlign.center,  
 ),  
 ),  
 ),  
 );  
 }  
}

**product\_overview\_screen.dart**

Where we show the list of product app barr and ProductGrid() function in child where this code in separate widget to organize the code.

import 'package:flutter/cupertino.dart';  
import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
import 'package:shop\_project/providers/cart.dart';  
import '../widgets/app\_drawer.dart';  
  
  
import '../widgets/products\_grid.dart';  
import '../widgets/badge.dart';  
import '../providers/cart.dart';  
import '../screens/cart\_screen.dart';  
  
enum FilterOption {  
 Favourites,  
 All,  
}  
  
class ProductsOverviewScreen extends StatefulWidget {  
  
 @override  
 \_ProductsOverviewScreenState createState() => \_ProductsOverviewScreenState();  
}  
  
class \_ProductsOverviewScreenState extends State<ProductsOverviewScreen> {  
 var \_showOnlyFavorites = false;  
  
 @override  
 Widget build(BuildContext context) {  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('MyShop'),  
 actions: <Widget>[  
 PopupMenuButton(  
 onSelected: (FilterOption selectedValue){  
 setState(() {  
 if (selectedValue == FilterOption.Favourites){  
 \_showOnlyFavorites = true;  
 }  
 else{  
 \_showOnlyFavorites = false;  
 }  
 });  
  
  
 },  
 icon: Icon(  
 Icons.*more\_vert*,  
 ),  
 itemBuilder: (\_) =>[  
 PopupMenuItem(child: Text('Only favouite'), value: FilterOption.Favourites),  
 PopupMenuItem(child: Text('Show all'), value: FilterOption.All,),  
 ],  
 ),  
 Consumer<Cart>(builder: (\_, cart, ch) => Badge(  
 child: ch ,  
 value: cart.itemCount.toString(),  
 ),  
 child: IconButton(  
 icon: Icon(  
 Icons.*shopping\_cart*,  
 ),  
 onPressed:(){  
 Navigator.*of*(context).pushNamed(CartScreen.*routeName*);  
 } ,  
 ),  
 ),  
 ],  
 ),  
 drawer: AppDrawer(),  
  
 body: ProductsGrid(\_showOnlyFavorites),  
  
  
  
 );  
 }

}

Product\_detail\_screen.dart

They display the detail of each product

import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
import 'package:shop\_project/providers/products.dart';  
import 'package:flutter\_rating\_bar/flutter\_rating\_bar.dart';  
  
  
  
class ProductDetailScreen extends StatelessWidget {  
// final String title;  
// ProductDetailScreen(this.title);  
static const *routeName* = '/product-detail';  
 @override  
 Widget build(BuildContext context) {  
 final productId = ModalRoute.*of*(context).settings.arguments as String;  
 final loadedProduct = Provider.*of*<Products>(context, listen:false).findById(productId);//this condition is use when you tap any product so the title of detail product set the names of product  
  
 return Scaffold(  
 appBar: AppBar(  
 title: Text(loadedProduct.title),  
 ),  
 body: SingleChildScrollView(  
 child: Column(  
 children: [  
 Container(  
 height: 300,  
 width: double.*infinity*,  
 child: Image.network(  
 loadedProduct.imageUrl,  
 fit: BoxFit.cover,  
 ),  
 ),  
 SizedBox(height: 10),  
 Text('\$${loadedProduct.price}',  
 style: TextStyle(  
 color: Colors.*grey*,  
 fontSize: 20,  
  
 ),  
 ),  
 SizedBox(height: 10),  
 Container(  
 padding: EdgeInsets.symmetric(horizontal: 10),  
 width: double.*infinity*,  
 child: Text(  
 loadedProduct.description,  
 textAlign: TextAlign.center,  
 softWrap: true,),  
 ),  
 SizedBox(height: 50,),  
 RatingBar.builder(  
 initialRating: 3,  
 minRating: 1,  
 direction: Axis.horizontal,  
 allowHalfRating: true,  
 itemCount: 5,  
 itemPadding: EdgeInsets.symmetric(horizontal: 4.0),  
 itemBuilder: (context, \_) => Icon(  
 Icons.*star*,  
 color: Colors.*amber*,  
 ),  
 onRatingUpdate: (rating) {  
 print(rating);  
 },  
 ),  
  
 ],  
 ),  
  
 )  
 );  
 }  
}

**Cart.data:**

import 'package:flutter/foundation.dart';  
  
class CartItem {  
 final String id;  
 final String title;  
 final int quantity;  
 final double price;  
  
 CartItem({  
 @required this.id,  
 @required this.title,  
 @required this.price,  
 @required this.quantity  
});  
  
  
}  
  
class Cart with ChangeNotifier{  
 Map<String, CartItem> \_items ={};  
  
 Map<String, CartItem> get items{  
 return {...\_items};  
 }  
int get itemCount{  
 return \_items.length;  
}  
  
double get totalAmount {  
 var total = 0.0;  
 \_items.forEach((key, cartItem) {  
 total += cartItem.price \* cartItem.quantity;  
 total.toStringAsFixed(2); // for my output will get the total with only two decimal numbers.  
 });  
 return total;  
  
}  
  
 void addItem(String productId, double price, String title,){  
 if(\_items.containsKey(productId)){  
 //change quantity  
 \_items.update(productId, (existingCartItem) => CartItem(  
 id: existingCartItem.id,  
 title: existingCartItem.title,  
 price: existingCartItem.price,  
 quantity: existingCartItem.quantity + 1 ,  
 ));  
 }  
 else{  
 \_items.putIfAbsent(productId, () => CartItem(  
 id: DateTime.now().toString(),  
 title: title,  
 price: price,  
 quantity: 1,  
 ),  
 );  
 }  
notifyListeners();  
 }  
  
 void removeItem(String productId){  
 \_items.remove(productId);  
 notifyListeners();  
 }  
 addOrRemoveQuantity(String productId, bool operators) {  
 if (\_items.containsKey(productId)) {  
 \_items.update(  
 productId,  
 (existingCartItem) => CartItem(  
 id: existingCartItem.id,  
 title: existingCartItem.title,  
 price: existingCartItem.price,  
 quantity: operators  
 ? existingCartItem.quantity + 1  
 : existingCartItem.quantity - 1),  
 );  
 }  
 notifyListeners();  
 }  
  
 void clear(){  
 //this condition when cart go to order section its clear cart  
 \_items = {};  
 notifyListeners();  
 }  
  
}

**cart\_item.dart:**

import 'package:flutter/cupertino.dart';  
import 'package:flutter/material.dart';  
import 'package:provider/provider.dart';  
import '../providers/cart.dart';  
  
  
class CartItem extends StatelessWidget {  
 final String id;  
 final String productId;  
 final double price;  
 final int quantity;  
 final String title;  
  
 CartItem(  
 this.id,  
 this.productId,  
 this.price,  
 this.quantity,  
 this.title);  
 @override  
 Widget build(BuildContext context) {  
 return Dismissible(  
 key: ValueKey(id),  
 background: Container(  
 color: Theme.*of*(context).errorColor,  
 child: Icon(  
 Icons.*delete*,  
 color: Colors.*white*,  
 size: 40,  
 ),  
  
  
 alignment: Alignment.*centerRight*,  
 padding: EdgeInsets.only(right: 20),  
 ) ,  
 direction: DismissDirection.endToStart,//right and left delete icon in dismissible  
 onDismissed: (direction){  
 Provider.*of*<Cart>(context, listen: false).removeItem(productId);  
 },  
  
 child: Card(  
 margin: EdgeInsets.symmetric(horizontal: 15,  
 vertical: 4),  
 child: Padding(  
 padding: EdgeInsets.all(8),  
 child: ListTile(  
 leading: CircleAvatar(  
 child: Padding(  
 padding: EdgeInsets.all(5),  
 child: FittedBox(  
 child: Text('\$$price'),),  
 ),  
 ),  
 title: Text(title),  
 subtitle: Text('Total: \$${(price \* quantity)}'),  
 trailing: Container(  
 child: Row(  
 mainAxisSize: MainAxisSize.min,  
 children: <Widget>[  
 IconButton(  
 onPressed: (quantity > 1)  
 ? () => {  
 Provider.*of*<Cart>(context, listen: false)  
 .addOrRemoveQuantity(productId, false)  
 }  
 : () => {  
 Provider.*of*<Cart>(context, listen: false)  
 .removeItem(productId)  
 },  
 icon: Icon(Icons.*remove*, color: Colors.*black*),  
 ),  
 Text('$quantity'),  
 IconButton(  
 icon: Icon(Icons.*add*, color: Colors.*black*),  
 onPressed: () {  
 Provider.*of*<Cart>(context, listen: false)  
 .addOrRemoveQuantity(productId, true);  
 },  
 ),  
  
 ],  
 ),  
 )  
  
 ),  
 ),  
 ),  
 );  
 }  
}

**cart\_screen.dart**

import 'package:flutter/material.dart';  
import 'package:shop\_project/providers/cart.dart';  
import '../providers/cart.dart';  
import 'package:provider/provider.dart';  
import '../widgets/cart\_item.dart' as ci ; //ci is use because CartItem is name mostly in dart file  
import '../providers/orders.dart';  
  
  
  
class CartScreen extends StatelessWidget {  
 static const *routeName* = '/cart';  
 @override  
 Widget build(BuildContext context) {  
 final cart = Provider.*of*<Cart>(context);  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('Your Cart'),  
 ),  
 body: Column(  
 children: <Widget> [  
 Card(  
 margin: EdgeInsets.all(15.0),  
 child: Padding(  
 padding: EdgeInsets.all(8),  
 child: Row(  
 mainAxisAlignment: MainAxisAlignment.spaceBetween,  
 children: <Widget> [  
 Text('Total', style: TextStyle(fontSize: 20),),  
 // SizedBox(width: 10.0),  
 Spacer(),  
 Chip(label: Text('\$${cart.totalAmount.toStringAsFixed(2)}' ,style: TextStyle(  
 color: Theme.*of*(context).primaryTextTheme.title.color  
 ),  
 ),  
 backgroundColor: Theme.*of*(context).primaryColor),  
 FlatButton(  
 child: Text('Order Now'),  
 onPressed: (){  
 Provider.*of*<Orders>(context, listen: false).addOrder(  
 cart.items.values.toList(),  
 cart.totalAmount,  
 );  
 cart.clear();  
 },  
 textColor: Theme.*of*(context).primaryColor,  
 )  
 ],  
 ),  
 ),  
 ),  
 SizedBox(height: 10),  
 Expanded(  
 child: ListView.builder(  
 itemCount: cart.items.length,  
 itemBuilder: (ctx,i) => ci.CartItem(  
 cart.items.values.toList()[i].id,  
 cart.items.keys.toList()[i],  
 cart.items.values.toList()[i]. price,  
 cart.items.values.toList()[i]. quantity,  
 cart.items.values.toList()[i]. title,  
  
 ),  
  
 ),  
 ),  
 ],  
 ),  
 );  
 }  
}

**Order.data**

import 'package:flutter/foundation.dart';  
  
import './cart.dart';  
  
class OrderItem {  
 final String id;  
 final double amount;  
 final List<CartItem> products;  
 final DateTime dateTime;  
  
  
 OrderItem({  
 @required this.id,  
 @required this.amount,  
 @required this.products,  
 @required this.dateTime,  
 });  
}  
class Orders with ChangeNotifier {  
 List<OrderItem> \_orders = [];  
 List<OrderItem> get orders {  
 return [...\_orders];  
 }  
void addOrder (List<CartItem> cartProducts, double total){  
 \_orders.insert(  
 0,  
 OrderItem(  
 id: DateTime.now().toString(),  
 amount: total,  
 dateTime: DateTime.now(),  
 products: cartProducts,  
 ),  
  
 );  
 notifyListeners();  
}  
  
}

**order\_item.dart**

import 'package:flutter/material.dart';  
import 'package:intl/intl.dart'; //we use intl package to use Dateformat  
import '../providers/orders.dart' as ord;  
import 'dart:math';  
  
class OrderItem extends StatefulWidget {  
 final ord.OrderItem order;  
 OrderItem(this.order);  
  
 @override  
 \_OrderItemState createState() => \_OrderItemState();  
}  
  
class \_OrderItemState extends State<OrderItem> {  
 var \_expanded = false;  
 @override  
 Widget build(BuildContext context) {  
 return Card(  
 margin: EdgeInsets.all(10),  
 child: Column(  
 children: <Widget>[  
 ListTile(  
 title: Text('\$${widget.order.amount}'),  
 subtitle: Text(DateFormat('dd/MM/yyyy hh:mm').format(widget.order.dateTime),  
 ),  
 trailing: IconButton(  
 icon: IconButton(  
 icon: Icon(  
 \_expanded ? Icons.*expand\_less* : Icons.*expand\_more* ),  
 onPressed: (){  
 setState(() {  
 \_expanded = ! \_expanded;  
 });  
 },  
 ),  
 ),  
 ),  
 if (\_expanded) Container(  
 padding: EdgeInsets.symmetric(horizontal: 15, vertical: 4),  
 height: min(widget.order.products.length \* 20.0 + 10, 100) , //for sizing when we press the epand button it show order items so we set these sizes  
 child: ListView(  
 children: widget.order.products.map((prod)=>  
 Row(  
 mainAxisAlignment: MainAxisAlignment.spaceBetween,  
 children:<Widget> [  
 Text(  
 prod.title,  
 style: TextStyle(  
 fontSize: 18,  
 fontWeight: FontWeight.*bold*,  
 ),  
 ),  
 Text(  
 '${prod.quantity}x \$${prod.price}',  
 style: TextStyle(  
 fontSize: 18,  
 color: Colors.*grey*,  
 ),  
 )  
 ],  
 )).toList(),  
 ),  
 )  
 ],  
 ),  
 );  
 }  
}

**Order\_screen.dart**

import 'package:flutter/material.dart';  
import '../widgets/app\_drawer.dart';  
import '../providers/orders.dart' show Orders;  
import 'package:provider/provider.dart';  
import '../widgets/order\_item.dart';  
  
  
class OrderScreen extends StatelessWidget {  
 static const *routeName* = '/orders';  
 @override  
 Widget build(BuildContext context) {  
 final orderData = Provider.*of*<Orders>(context);  
 return Scaffold(  
 appBar: AppBar(  
 title: Text('Your orders'),  
 ),  
 drawer: AppDrawer(),  
 body: ListView.builder(  
 itemCount: orderData.orders.length,  
 itemBuilder:(ctx,i)=> OrderItem(orderData.orders[i]) ,  
 ),  
 );  
 }  
}